

Resumen

Este Proyecto Fin de Carrera trata sobre el reconocimiento e identificación de caracteres de matrículas de automóviles. Este tipo de sistemas de reconocimiento también se los conoce mundialmente como sistemas ANPR ("Automatic Number Plate Recognition") o LPR ("License Plate Recognition"). La gran cantidad de vehículos y logística que se mueve cada segundo por todo el planeta, hace necesaria su registro para su tratamiento y control. Por ello, es necesario implementar un sistema que pueda identificar correctamente estos recursos, para su posterior procesamiento, construyendo así una herramienta útil, ágil y dinámica.

El presente trabajo ha sido estructurado en varias partes. La primera de ellas nos muestra los objetivos y las motivaciones que se persiguen con la realización de este proyecto.

En la segunda, se abordan y desarrollan todos los diferentes procesos teóricos y técnicos, así como matemáticos, que forman un sistema ANPR común, con el fin de implementar una aplicación práctica que pueda demostrar la utilidad de estos en cualquier situación.

En la tercera, se desarrolla esa parte práctica en la que se apoya la base teórica del trabajo. En ésta se describen y desarrollan los diversos algoritmos, creados con el fin de estudiar y comprobar todo lo planteado hasta ahora, así como observar su comportamiento. Se implementan varios procesos característicos del reconocimiento de caracteres y patrones, como la detección de áreas o patrones, rotado y transformación de imágenes, procesos de detección de bordes, segmentación de caracteres y patrones, umbralización y normalización, extracción de características y patrones, redes neuronales, y finalmente el reconocimiento óptico de caracteres o comúnmente conocido como OCR.

La última parte refleja los resultados obtenidos a partir del sistema de reconocimiento de caracteres implementado para el trabajo y se exponen las conclusiones extraídas a partir de éste. Finalmente se plantean las líneas futuras de mejora, desarrollo e investigación, para poder realizar un sistema más eficiente y global.

Abstract

This Thesis deals about license plate characters recognition and identification. These kinds of systems are also known worldwide as ANPR systems ("Automatic Number Plate Recognition") or LPR ("License Plate Recognition"). The great number of vehicles and logistics moving every second all over the world, requires a registration for treatment and control. Thereby, it's therefore necessary to implement a system that can identify correctly these resources, for further processing, thus building a useful, flexible and dynamic tool.

This work has been structured into several parts. The first one shows the objectives and motivations attained by the completion of this project.

In the second part, it's developed all the different theoretical and technical processes, forming a common ANPR system in order to implement a practical application that can demonstrate the usefulness of these ones on any situation.

In the third, the practical part is developed, which is based on the theoretical work. In this one are described and developed various algorithms, created to study and verify all the questions until now suggested, and complain the behavior of these systems. Several recognition of characters and patterns characteristic processes are implemented, such as areas or patterns detection, image rotation and transformation, edge detection processes, patterns and character segmentation, thresholding and normalization, features and patterns extraction, neural networks, and finally the optical character recognition or commonly known like OCR.

The last part shows the results obtained from the character recognition system implemented for this thesis and the outlines conclusions drawn from it. Finally, future lines of improvement, research and development are proposed, in order to make a more efficient and comprehensive system.



UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA UNIVERSITARIA DE
INGENIERÍA TÉCNICA DE
TELECOMUNICACIÓN

IDENTIFICACIÓN Y RECONOCIMIENTO DE MATRÍCULAS DE AUTOMÓVILES CON MATLAB

AUTOR: SAMUEL ENRIQUE CEBALLOS JIMÉNEZ

DIRECTOR: D. ALFONSO MARÍN MARCOS

Año: 2012

Agradecimientos

Primeramente, agradecer a mi tutor, Alfonso Martín Marcos, por darme la oportunidad de llevar a cabo el PFC con él, y haber aprendido tanto en las asignaturas de Sistemas de Televisión y Tratamiento Digital de la Imagen.

A todos mis compañeros de primer año, especialidad, y en especial a los del viaje de fin de curso, por esos momentos inolvidables.

A mi amigo de carrera Miguel, de trabajo Jorge y mi amigo Sergio, por la ayuda y ánimo prestado para la creación y confección a lo largo de este proyecto. A mi compañero de trabajo Gregorio, por la cantidad de ayuda mutua recibida y que ha desembocado en la creación de dos proyectos brillantes. A mi amigo de toda la vida Víctor, por un camino en el que empezamos y vamos a acabar juntos.

A mi novia M^aCarmen, por el tan largo camino que hemos recorrido juntos, por tantas tardes y noches de estudio, y las que nos quedan por volver a repetir. Por tu apoyo y tus ánimos durante este duro trayecto, que van a ser los mismos que te voy a proporcionar para llegar a ser la grandísima profesional que eres. Que sepas que siempre estaré ahí. Te quiero.

A mis abuelos, que siempre han tenido el sueño de tener un ingeniero en la familia y a su vez a alguien por el que estar tremendamente orgullosos.

Por último a mis padres, los verdaderos productores de esta travesía, los cuales han dado todo lo posible por que esté ahora mismo aquí, y me han dado una oportunidad que ellos en su día jamás la pudieron tener.

Gracias a todos

Índice

| | |
|---|----|
| Agradecimientos | 3 |
| Índice de Figuras..... | 10 |
| Índice de Tablas..... | 14 |
| Resumen | 15 |
| 1 Introducción | 16 |
| 1.1 Introducción | 17 |
| 1.2 Motivación y objetivos..... | 17 |
| 1.3 Organización del proyecto | 18 |
| 2 Matrículas automovilísticas en España | 21 |
| 2.1 Historia..... | 22 |
| 2.1.1 Sistema provincial numérico..... | 22 |
| 2.1.2 Sistema provincial alfanumérico | 23 |
| 2.1.3 Sistema nacional..... | 23 |
| 2.2 Otras matrículas..... | 26 |
| 2.2.1 Las matrículas especiales..... | 26 |
| 2.2.2 Otras | 27 |
| 3 Reconocimiento Automático de Matrículas ANPR..... | 29 |
| 3.1 Introducción | 30 |

| | | |
|------------|--|----|
| 3.2 | Aspectos físico-técnicos | 31 |
| 3.3 | Funcionamiento | 33 |
| 3.4 | Aplicaciones | 35 |
| 3.5 | Los componentes del sistema | 36 |
| 3.5.1 | Enfoque del proyecto | 38 |
| 3.5.2 | Procesos y subprocesos del proyecto..... | 40 |
| 3.6 | Aspectos matemáticos..... | 40 |
| 3.7 | Dificultades..... | 41 |
| 4 | Principios de detección de áreas..... | 44 |
| 4.1 | Detección de bordes y rango de filtrado | 45 |
| 4.1.1 | Convolución de matrices | 46 |
| 4.2 | Proyección horizontal y vertical de la imagen | 50 |
| 4.3 | Análisis estadístico de doble fase | 51 |
| 4.3.1 | La Instantánea | 52 |
| 4.3.2 | La franja..... | 52 |
| 4.3.3 | La placa | 52 |
| 4.3.4 | Detección vertical – recorte de franja..... | 53 |
| 4.3.5 | Detección horizontal – recorte de placa..... | 55 |
| 4.4 | Análisis heurístico y prioridad de selección de candidatos | 57 |

| | | |
|------------|--|----|
| 4.4.1 | Selección de prioridades y análisis heurístico de bandas | 58 |
| 4.4.2 | Análisis heurístico profundo | 59 |
| 4.5 | Mecanismo de rotación..... | 61 |
| 4.5.1 | Detección de la desviación | 61 |
| 4.5.2 | Corrección de la desviación | 64 |
| 4.5.3 | La transformación proyectiva..... | 65 |
| 5 | Principios de segmentación de caracteres..... | 69 |
| 5.1 | Segmentación de la placa usando una proyección horizontal..... | 70 |
| 5.2 | Extracción de los caracteres de los segmentos horizontales..... | 72 |
| 5.2.1 | Extracción de las piezas | 73 |
| 5.2.2 | Análisis heurístico de las piezas | 74 |
| 6 | Extracción de características y normalización de caracteres..... | 76 |
| 6.1 | Normalización del brillo y el contraste | 77 |
| 6.1.1 | Normalización del histograma | 77 |
| 6.1.2 | Umbralización global..... | 79 |
| 6.1.3 | Umbralización adaptativa | 81 |
| 6.2 | Normalización de las dimensiones y remuestreo..... | 82 |
| 6.2.1 | Diezmado de vecindad más cercana..... | 83 |
| 6.2.2 | Diezmado promedio ponderado | 84 |

| | | |
|------------|--|-----|
| 6.3 | Extracción de características | 85 |
| 6.3.1 | La matriz de píxeles | 85 |
| 6.3.2 | Detección de bordes del carácter | 86 |
| 6.3.3 | Esqueletización y análisis estructural | 89 |
| 7 | Reconocimiento óptico de caracteres OCR | 97 |
| 7.1 | Clasificación del problema general | 98 |
| 7.2 | La neurona biológica y sus modelos matemáticos | 100 |
| 7.2.1 | La neurona de McCulloch-Pitts | 102 |
| 7.2.2 | Perceptrón | 102 |
| 7.3 | Red neuronal prealimentada | 103 |
| 7.4 | Adaptación al mecanismo de red neuronal prealimentado | 105 |
| 7.4.1 | Fase activa | 107 |
| 7.4.2 | Derivadas parciales y gradiente de la función de error | 107 |
| 7.4.3 | Fase de adaptación | 108 |
| 7.5 | Análisis heurístico de caracteres | 110 |
| 8 | Análisis sintáctico de la placa | 114 |
| 8.1 | Principios y algoritmos | 115 |
| 8.1.1 | Reconocimiento del carácter y su coste | 115 |
| 8.1.2 | Patrones sintácticos | 116 |

| | | |
|------------|---|-----|
| 8.1.3 | Elección del patrón correcto..... | 116 |
| 9 | Pruebas y consideraciones finales..... | 118 |
| 9.1 | Elección del conjunto representativo de imágenes..... | 119 |
| 9.2 | Evaluación de la exactitud de una matrícula | 120 |
| 9.2.1 | Puntuación binaria | 121 |
| 9.2.2 | Puntuación ponderada..... | 121 |
| 9.2.3 | Porcentaje..... | 122 |
| 9.3 | Resultados..... | 122 |
| 10 | Conclusiones y líneas futuras | 124 |
| 9.4 | Cumplimiento de objetivos y conclusiones | 125 |
| 9.5 | Líneas futuras..... | 128 |
| | Bibliografía..... | 129 |
| | Apéndice I: Notaciones y símbolos matemáticos..... | 132 |
| | Apéndice II: Contenido del CD..... | 135 |

Índice de Figuras

| | |
|---|----|
| Figura 1. Ejemplo de matrícula provincial numérica. | 22 |
| Figura 2. Ejemplo y variaciones de una matrícula provincial alfanumérica para la provincia de Álava. | 23 |
| Figura 3. Matrícula ordenada actual y más utilizada en nuestro país. | 26 |
| Figura 4. Distintos tipos de matrículas especiales. | 27 |
| Figura 5. Equipo de una estación de control móvil. | 30 |
| Figura 6. Diagrama de bloques de un sistema ANPR en un aparcamiento. ... | 31 |
| Figura 7. El vehículo llega a la entrada, y el lazo inductivo da una señal al PC de control. | 34 |
| Figura 8. Creación de la imagen digital del vehículo. | 34 |
| Figura 9. Lectura de la matrícula de la imagen digitalizada. | 34 |
| Figura 10. Permiso de acceso y almacenamiento de los datos en el historial. | 35 |
| Figura 11. Procesos del sistema. | 37 |
| Figura 12. Nivel de extracción y procesado de datos. | 38 |
| Figura 13. Flujo de trabajo del sistema. | 39 |
| Figura 14. Ejemplo de matrículas conflictivas por diversos factores. | 41 |
| Figura 15. (a) Efectos de los materiales reflexivos en una matrícula y (b) Efecto de motion blur. | 42 |
| Figura 16. El píxel está afectado por sus vecinos conforme a la matriz de convolución. | 46 |
| Figura 17. Filtrado Sobel horizontal y vertical. | 47 |
| Figura 18. Filtrados Sobel y Canny respectivamente. | 49 |

| | |
|--|----|
| Figura 19. Varios tipos de filtrados promedios y de detección de bordes direccionales. | 50 |
| Figura 20. Proyección vertical de una imagen en el eje y..... | 51 |
| Figura 21. Recorte de placa de dos fases. | 53 |
| Figura 22. Proyección vertical de la imagen convolucionada con una matriz de promedio. | 54 |
| Figura 23. Franja detectada por el análisis de la proyección vertical. | 54 |
| Figura 24. Proyección horizontal $px(x)$ y su derivada $px'(x)$ | 56 |
| Figura 25. Área mayor de la matrícula detectada después del proceso de rotación..... | 57 |
| Figura 26. Segmentación en caracteres individuales de la matrícula para un análisis heurístico más profundo. | 60 |
| Figura 27. Histograma de brillo de la matrícula..... | 60 |
| Figura 28. (a) Matrícula capturada sobre un ángulo correcto (b) Placa girada o rotada (c) Placa distorsionada o torcida..... | 61 |
| Figura 29. Sistemas de coordenadas "XY" y "AB". | 62 |
| Figura 30. Proceso gráfico de la transformada de Hough. | 63 |
| Figura 31. (a) Matrícula original y (b) Matrícula después del proceso de enderezado. | 65 |
| Figura 32. Principio de la transformación proyectiva. | 66 |
| Figura 33. Ejemplos de transformación proyectiva. | 67 |
| Figura 34. Transformación proyectiva aplicada a una matrícula. | 68 |
| Figura 35. Ejemplo de transformación proyectiva. | 68 |

| | |
|--|----|
| Figura 36. (a) Matrícula después de la aplicación de un umbral adaptativo. (b) Proyección horizontal de la placa con los picos detectados. | 71 |
| Figura 37. El segmento horizontal de la matrícula contiene varios grupos (piezas) de píxeles vecinos. | 73 |
| Figura 38. Ejemplo de conexión de 4 píxeles u 8 píxeles..... | 74 |
| Figura 39. Imagen de entrada y salida después de la fase de segmentación. | 75 |
| Figura 40. Utilizamos una interpolación polinomial de Lagrange como una función de transformación. | 79 |
| Figura 41. Resultado erróneo de una umbralización global..... | 80 |
| Figura 42. Matrícula procesada por la aproximación de Chow y Kaneko. | 81 |
| Figura 43. Un píxel en la imagen remuestreada corresponde a un grupo de píxeles en la imagen original. | 83 |
| Figura 44. Resultado a partir del método de <i>diezmado de pixel vecino más cercano</i> y por <i>diezmado promedio ponderado</i> respectivamente. | 84 |
| Figura 45. Ejemplo de matriz de píxeles para la extracción de características. | 86 |
| Figura 46. Diseños de las regiones en el mapa de bits del carácter. | 87 |
| Figura 47. El mapa de bits procesado puede contener diferentes tipos de bordes de dimensiones 2x2. | 87 |
| Figura 48. Conjunto de píxeles interiores y límite del carácter..... | 90 |
| Figura 49. Mapa de bits del carácter durante el proceso de esqueletización. | 92 |
| Figura 50. Ejemplos de unión, fin de línea y bucle respectivamente. | 93 |
| Figura 51. Ilustración y proceso del término “lago” o “lazo”. | 94 |

| | |
|--|-----|
| Figura 52. (a) Invariabilidad anterior o posterior de la aplicación de las limitaciones estructurales y (b) ejemplo de alfabeto después de la esqueletización..... | 95 |
| Figura 53. Ilustración del proceso de extracción de características..... | 96 |
| Figura 54. Proyección entre los conjuntos A y B. | 99 |
| Figura 55. La neurona biológica..... | 101 |
| Figura 56. (a) Ilustración esquemática de una célula neuronal y (b) Conexión sináptica entre una dendrita y un botón terminal del axón. | 101 |
| Figura 57. (a) Función del sumatorio y la ganancia (saturación) g del perceptrón con un umbral implementado como un valor de entrada dedicado. (b) Función de saturación sigmoidea..... | 103 |
| Figura 58. Arquitectura de una red neuronal prealimentada de tres capas. | 105 |
| Figura 59. Dependencia de las funciones de error E_t y E_x en el número de neuronas en la capa de entrada (n) y en el número de pasos iterativos (k). | 106 |
| Figura 60. Aproximación numérica de la búsqueda del mínimo global en el “landscape error” | 109 |
| Figura 61. Caracteres segmentados antes (a) y después (b) de aplicar el algoritmo de extracción. | 111 |
| Figura 62. Ejemplos de imagen del conjunto de matrículas (a) claras (b) con pequeños o caracteres borrosos (c) matrículas distorsionadas (d) placas con entornos conflictivos o sombras parciales..... | 120 |

Índice de Tablas

| | |
|--|-----|
| Tabla 1. Áreas del sistema tratadas en el proyecto. | 40 |
| Tabla 2. Heurísticas utilizadas en el proceso de selección de candidatos. | 59 |
| Tabla 3. Limitaciones estructurales de todos los caracteres que pueden ser utilizados..... | 95 |
| Tabla 4. Propiedades de los segmentos de la figura 61..... | 113 |
| Tabla 5. Tasas de reconocimiento del sistema ANPR según el tipo de matrícula. | 122 |
| Tabla 6. Tasa de reconocimiento del sistema LPR global. | 122 |

Resumen

Este Proyecto Fin de Carrera trata sobre el reconocimiento e identificación de caracteres de matrículas de automóviles. Este tipo de sistemas de reconocimiento también se los conoce mundialmente como sistemas ANPR ("Automatic Number Plate Recognition") o LPR ("License Plate Recognition"). La gran cantidad de vehículos y logística que se mueve cada segundo por todo el planeta, hace necesaria su registro para su tratamiento y control. Por ello, es necesario implementar un sistema que pueda identificar correctamente estos recursos, para su posterior procesamiento, construyendo así una herramienta útil, ágil y dinámica.

El presente trabajo ha sido estructurado en varias partes. La primera de ellas nos muestra los objetivos y las motivaciones que se persiguen con la realización de este proyecto.

En la segunda, se abordan y desarrollan todos los diferentes procesos teóricos y técnicos, así como matemáticos, que forman un sistema ANPR común, con el fin de implementar una aplicación práctica que pueda demostrar la utilidad de estos en cualquier situación.

En la tercera, se desarrolla esa parte práctica en la que se apoya la base teórica del trabajo. En ésta se describen y desarrollan los diversos algoritmos, creados con el fin de estudiar y comprobar todo lo planteado hasta ahora, así como observar su comportamiento. Se implementan varios procesos característicos del reconocimiento de caracteres y patrones, como la detección de áreas o patrones, rotado y transformación de imágenes, procesos de detección de bordes, segmentación de caracteres y patrones, umbralización y normalización, extracción de características y patrones, redes neuronales, y finalmente el reconocimiento óptico de caracteres o comúnmente conocido como OCR.

La última parte refleja los resultados obtenidos a partir del sistema de reconocimiento de caracteres implementado para el trabajo y se exponen las conclusiones extraídas a partir de éste. Finalmente se plantean las líneas futuras de mejora, desarrollo e investigación, para poder realizar un sistema más eficiente y global.

1

Introducción

1.1 Introducción

En la actualidad existe una gran cantidad de elementos o objetos que están numerados o poseen un número de serie como una característica propia de identificación. Manejar tal cantidad de información sería imposible a partir de los métodos de registro tradicionales. Por ello, se ideó una forma de digitalizar visualmente estos números de identificación para poder llevar un procesamiento más rápido y eficaz. A partir de esta idea nació el concepto del reconocimiento de patrones, en los que están basados los sistemas ANPR.

El diseño de los sistemas ANPR es un campo de investigación que abarca la inteligencia artificial, visión máquina, reconocimiento de patrones y de redes neuronales. Por ello, el principal objetivo de este proyecto es el estudio algorítmico y los principios matemáticos de los sistemas de reconocimiento automático de matrículas, así como realizar una prueba práctica para comprobar y verificar estos principios.

1.2 Motivación y objetivos

La principal motivación es proporcionar un marco de trabajo que permita comprender y estudiar el comportamiento tanto teórico como práctico de los sistemas ANPR, así como la contribución en el avance de la investigación en este campo. A continuación se muestran los objetivos pretendidos en este proyecto:

- Conocer los principios del reconocimiento de patrones y los sistemas en los que deriva.
- Conocer la estructura y características de una matrícula para su procesamiento.
- Estudiar el funcionamiento de un sistema ANPR completo, tanto la parte hardware como la de software.
- Exponer y desarrollar todas las partes y procesos que forman la parte software del sistema ANPR.
- Realizar una parte práctica, para justificar los principios teóricos desarrollados.
- Intentar mejorar o incluir partes o procedimientos que mejoren el sistema, así como implementar diseños propios.
- Realizar una fase de pruebas en los que comprobar los resultados.

- Obtener conclusiones de los algoritmos diseñados y plantear ideas con el fin de mejorarlo.

Cabe destacar el deseo de poder usar este proyecto como una introducción o un mara de salida hacia un ámbito más desarrollado y optimizado de los sistemas ANPR.

1.3 Organización del proyecto

El proyecto se ha organizado en los siguientes puntos enumerados:

- **Capítulo 1: Introducción.** En este capítulo se realiza una introducción las motivaciones que han llevados a la realización del trabajo y su necesidad de progreso. También se detalla la planificación del proyecto.
- **Capítulo 2: Matrículas automovilísticas en España.** Se describen las principales características de las matrículas en nuestro país y así poder abordar la parte práctica del proyecto de una forma útil y eficaz a partir de las mismas.
- **Capítulo 3: Reconocimiento Automático de Matrículas ANPR.** Detalla toda la tecnología y procedimientos utilizados en los sistemas ANPR actuales, así como todos los campos de acción en los que se utiliza.
- **Capítulo 4: Principios de detección de áreas.** Este capítulo trata de la dificultad en la detección del área de la matrícula. Esta problemática incluye algoritmos que son capaces de detectar el área rectangular de la matrícula en la imagen original. Las personas definirían una matrícula en un lenguaje natural como "una pequeña placa de una determinada forma hecha de plástico o metal adherida a un vehículo para fines identificativos", pero las máquinas no comprenden esta definición. Por ello, se necesita encontrar una definición alternativa basada en descripciones, las cuales deben ser comprendidas por las máquinas. Este es el problema fundamental de la visión máquina y de esta capítulo en concreto.
- **Capítulo 5: Principios de segmentación de caracteres.** Se describen los principios de la segmentación de caracteres. En la mayoría de los casos, la caracteres son segmentados usando la proyección horizontal de una matrícula preprocesada, pero a veces estos principios pueden fallar, especialmente si las matrículas detectadas están demasiado rotadas y transformadas

perspectivamente, por lo que se deben usar algoritmos de segmentación más sofisticados o completamente alternativos.

- **Capítulo 6: Extracción de características y normalización de caracteres.** Se tratan varios métodos de detección y normalización de caracteres. Primeramente la dimensiones y el brillo deben de ser normalizados para asegurar una invariabilidad en las condiciones de tamaño y color, por lo que debe establecerse un algoritmo de extracción que tenga la característica de ser aplicado sobre un carácter y filtre los datos irrelevantes. Es necesario extraer características, la cuales sean invariables sobre deformaciones, estilo de fuente usada, etc.
- **Capítulo 7: Reconocimiento óptico de caracteres OCR.** Se estudia la clasificación de patrones y las redes neuronales, además de su uso en el reconocimiento de parámetros. Los caracteres pueden ser clasificados y reorganizados por un simple algoritmo de vecinas cercanas aplicado a un vector de características extraídas, o existe también la posibilidad de usar uno de los más sofisticados métodos de clasificación, como el feed-forward (en español prealimentación) o el Hopfield neural networks. En este capítulo también se muestran análisis heurísticos y morfológicos adicionales, que son usados para la eliminación de los elementos que no sean caracteres de la matrícula.
- **Capítulo 8: Análisis sintáctico de la placa.** A veces el proceso de reconocimiento puede fallar y la matrícula reconocida presenta errores. Algunos de estos errores pueden ser detectados a través de un análisis sintáctico de la matrícula obtenida. Si obtenemos una expresión normal, o una regla de cómo evaluar una matrícula específica de un país, podremos reconstruir matrículas defectuosas usando esta regla. Por ejemplo, un número "0" puede ser automáticamente reparado por una "O" en ciertas posiciones, donde los números no son permitidos.
- **Capítulo 9: Pruebas y consideraciones finales.** Incluye los resultados finales de las pruebas llevadas a cabo en la parte práctica.
- **Capítulo 10: Conclusiones y líneas futuras.** Se exponen las conclusiones que han alcanzado tras la realización del proyecto. También se ofrece una descripción de las ampliaciones y mejoras posibles a realizar en la aplicación a corto-medio-plazo, además de nuevas e interesantes posibilidades que podrán incorporarse en un futuro más lejano

- **Bibliografía:** Detalla las fuentes de información usadas en la elaboración del presente proyecto
- **Apéndice I: Notaciones y símbolos matemáticos.** Muestra detalladamente las expresiones matemáticas utilizadas en la explicación de los distintos apartados del proyecto.
- **Apéndice II: Contenido del CD.** Ofrece una explicación de las partes y algoritmos creados en MATLAB, para la realización de la parte teórica.

2 Matrículas automovilísticas en España

Los vehículos españoles empezaron a matricularse en el año 1900. Desde entonces ha habido tres sistemas: el provincial numérico, el provincial alfanumérico y el nacional. El primero fue usado desde 1900 hasta 1971. El segundo, desde este año hasta septiembre del año 2000, cuando fue introducido el tercero. Los dos primeros eran a nivel provincial, y el tercero, a nivel nacional.

2.1 Historia

2.1.1 Sistema provincial numérico

Desde la implantación de las matrículas en España el 31 de octubre del año 1900, las placas han sido blancas con los números negros; incluían en primer lugar una o dos (A, AB, CC, CS, P, SG, TE,...) o incluso tres letras al principio (ALB para Albacete, CAC para Cáceres, etc.) que representaban a una provincia (Alicante, Albacete, Cáceres, Castellón, Palencia, Segovia, Teruel,...), y cifras, hasta 6, de esta manera: XX-000000 (sin ceros a la izquierda).



Figura 1. Ejemplo de matrícula provincial numérica.

Cuando las iniciales de los nombres de las provincias presentaban un parecido notable con las de otra, se utilizaba una sola letra para la provincia de mayor población y dos letras para las de menor (**A** Alicante, frente a **AL** Almería; **B** Barcelona, frente a **BA** Badajoz; **C** Coruña, frente a **CO** Córdoba; **H** Huelva, frente a **HU** Huesca; **L** Lérida, frente a **LE** León; **M** Madrid, frente a **MA** Málaga; **S** Santander, frente a **SA** Salamanca; **V** Valencia, frente a **VA** Valladolid; **Z** Zaragoza, frente a **ZA** Zamora). Si la capital era heterónoma con respecto a la provincia, se prefirieron las siglas de la capital (**PM** en lugar de Baleares; **VI** en lugar de Álava; **BI** en lugar de Vizcaya; **SS** en lugar de Guipúzcoa; **O** en lugar de Asturias) excepto en el caso de **NA**, donde el topónimo Pamplona no se impuso

y en Canarias, donde lo que prevaleció fue el nombre de la isla mayor de cada provincia, **GC** y **TF**. En el resto de los territorios donde podía haber confusión, la segunda letra era la diferenciadora (**CC** Cáceres, **CA** Cádiz, **CE** Ceuta, **CU** Cuenca; **SE** Sevilla, **SG** Segovia, **SO** Soria, etc.).

En los últimos años en que estuvo vigente el uso de siglas provinciales, la Dirección General de Tráfico permitió, en el ámbito de la normalización lingüística con el resto de las lenguas cooficiales del Estado, que tanto las placas ya instaladas como las de nueva creación de tres provincias pudieran ser modificadas así: **GE** Gerona como **GI** Girona; **OR** Orense como **OU** Ourense; **PM** Baleares como **IB** Illes Balears.

2.1.2 Sistema provincial alfanumérico

En el año 1971, se reinició la numeración añadiéndose una letra y reduciéndose las cifras a cuatro, XX-0000-A, y así sucesivamente hasta llegar a la Z; entonces se añadió una segunda letra, XX-0000-ZZ. Antes de agotar el sistema con la matrícula M-9999-ZZ (se llegó hasta M-6814-ZX), se pasaría al último sistema.

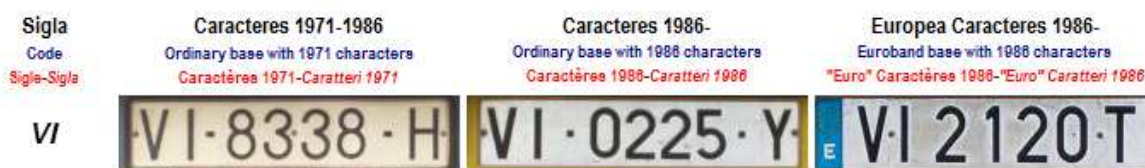


Figura 2. Ejemplo y variaciones de una matrícula provincial alfanumérica para la provincia de Álava.

2.1.3 Sistema nacional

El 17 de septiembre del año 2000 entró en vigor el nuevo sistema, introduciendo matrículas que constan de cuatro dígitos y tres letras consonantes (suprimiéndose las cinco vocales, y las letras **Ñ**, **Q**, **CH** y **LL**), **[E] 0000-BBB**, que se asignan a nivel nacional, de forma que ya no se puede conocer por la matrícula de qué provincia es el vehículo. El vehículo conserva la matrícula hasta su desguace y no se vuelve a asignar a otro vehículo.

2.1.2.1 Antecedentes

El 18 de septiembre del 2000, entró en vigor un nuevo sistema de matriculación de vehículos en España. Es el llamado modelo 'europeo', sin distintivos provinciales y con la "E" de España sobre la bandera de la Unión Europea.

A mediados de la década de los noventa, con el gobierno socialista, ya se planteó la implantación de un modelo de matrícula que incluyera la banda azul europea y eliminara los distintivos provinciales, sin embargo, las presiones de los partidos nacionalistas, obligaron a posponer la cuestión; ya que recomendaban que se estudiara un modelo con la inclusión de las siglas de las autonomías, en una solución que en nada satisfacía las pretensiones del sector comercial del automóvil.

A raíz de la mayoría absoluta obtenida por la derecha, se vuelve a abordar el problema, aunque no se llega a tomar ninguna decisión sobre la sustitución del sistema ordinario de matriculación, mientras que en alguna de las 52 Jefaturas Provinciales de Tráfico no se agotaran todas las combinaciones. Se elabora el nuevo Reglamento General de Vehículos (R.D. 2822/1998 de 23 de diciembre) en el que se introducen sustanciales novedades. Cambian todos los sistemas de matriculación especial a sistemas sin siglas provinciales con combinaciones de 4 números y 3 letras, la matriculación de motocicletas pasa a ser competencia del Estado y se permite la rematriculación de vehículos usados en provincias distintas a las de la matrícula original. Este reglamento entra en vigor el 27 de junio de 1999.

Cuando la Jefatura Provincial de Tráfico de Madrid comienza a otorgar la serie Z (el 31 de marzo se otorga M-3481-ZB), comienza una presión de la opinión pública y de los medios de comunicación, incitando al gobierno del Estado a que tome una decisión, puesto que se entiende que el sistema está a punto de agotarse en Madrid.

El 31 de agosto de 2000, el gobierno se pronuncia por primera vez, indicando que se desestima el modelo con la banda europea, y se continúa con el modelo de siglas provinciales, de modo que la provincia que agote el sistema de dos letras de serie, añadiría una tercera (así, después de M-9999-ZZ, iría M-0000-BBB), aludiendo que se trata de un modelo muy arraigado en España, con 100 años de existencia, con el que se identifica la sociedad española.

Esto desató numerosas críticas, sobre todo del sector comercial del automóvil, puesto que perjudica las expectativas del mercado de segunda mano. Los fabricantes de matrículas también se pronuncian puesto que han hecho una gran inversión al adecuar su maquinaria a la placa europea. Sectores políticos del partido en el poder, ven peligrar la imagen de nación única y homogénea, de no aplicarse el sistema europeo sin siglas provinciales.

Unos y otros proponen varias alternativas para el nuevo sistema, hasta que el día 8 de septiembre, el gobierno se vuelve a pronunciar diciendo que el

modelo europeo sin siglas provinciales y el modelo que está en vigor tienen el 50% de posibilidades. Se abre un gran debate en la opinión pública.

Por fin, el 15 de septiembre de 2000, se aprueba la Orden Modificativa del Reglamento General de Vehículos, en la que se rechaza el modelo provincial, y se adopta el modelo europeo sin sigla provincial, que entraría en vigor el 18 de septiembre de 2000, en el que no se llegaría a agotar el sistema vigente. Ni que decir tiene que ningún efecto tuvieron las alegaciones de la oposición ni los intentos de desobediencia civil propugnados desde algunos sectores de la población.

2.1.2.2 El nuevo sistema, razones oficiales para su implantación

Las razones oficiales dadas para la elección de este modelo son varias. La primera es contribuir “a la sensibilización ciudadana a favor de la integración europea”. También se responde, con ello, a un deseo de una parte de los españoles de que desaparezcan los distintivos provinciales y se facilitan las ventas de los coches de segunda mano entre diferentes provincias. Además, con ella, los coches que viajen a la Unión Europea no necesitan colocar el óvalo blanco con la “E” inscrita, aunque sí lo deben hacer si viajan por terceros países.

Todo ello contribuyó, además, a la sensibilización ciudadana a favor de la integración europea y a la identificación de los vehículos como pertenecientes a un mismo espacio europeo común. El modelo de placa de matrícula elegido es, además, el incorporado ya a sus respectivos sistemas jurídicos por la mayoría de los Estados miembros de la Unión Europea.

2.1.2.3 Descripción de la placa y del sistema de matriculación

El tamaño, de **52x11 cm**, es dos centímetros más largo que el modelo anterior, no incluye el distintivo provincial sino la “E” de España sobre la bandera de la Unión Europea más una combinación de cuatro números (de 0000 a 9999) y tres letras: comienza por BBB y termina en ZZZ y excluye las vocales (para evitar combinaciones malsonantes y acrósticos significativos) y la LL, CH (incompatibles con el diseño de la placa, que no admite cuatro caracteres en el último grupo) y la Ñ y Q, por confundirse con la N y O y el número 0.

Las letras utilizadas, por tanto, son la B,C,D,F,G,H,J,K,L,M,N,P,R,S,T,V,W,X,Y y Z, en orden alfabético. Se utilizan todos los números, formando 10.000 combinaciones de números de cuatro cifras. Para los números menores de

1000, se emplean los ceros a la izquierda: el 0 es el 0000, el 9 es el 0009, el 99 es el 0099 y el 999 es el 0999.

Este sistema permite $10000 \times 20 \times 20 \times 20 = 80.000.000$ de matriculaciones, lo que, al ritmo actual del país (alrededor de 2.000.000 vehículos anuales), mantendría útil el sistema durante algo más de 40 años.

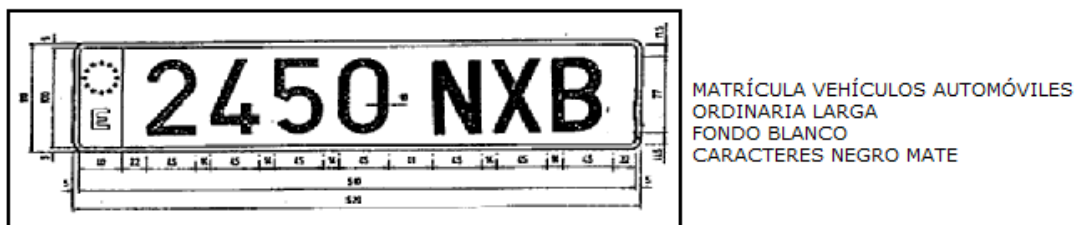


Figura 3. Matrícula ordinada actual y más utilizada en nuestro país.

El nuevo reparto de números de placa de matrícula se realiza por la unidad central de informática. Cada Jefatura Provincial solicita un cupo de matrículas para cubrir sus necesidades en un periodo de tiempo corto, unos días, una semana. Cuando éstas se agotan solicita el siguiente cupo. Entre petición y petición es muy probable que otras Jefaturas Provinciales hayan solicitado sus cupos, por lo que cada provincia dispondrá al mes de varios cupos, que no necesariamente serán correlativos entre sí.

Los titulares de vehículos matriculados con anterioridad a la entrada en vigor de la Orden de 15/09/2000, podrán sustituir, con carácter voluntario, las placas de matrícula por otras en las que figure la bandera comunitaria, cuyas dimensiones se ajustarán a las nuevas previstas, si bien conservarán el número de matrícula que tuvieran asignado. En el caso de que se vean obligados a sustituir las placas de matrícula por pérdida, sustracción o deterioro, deberán necesariamente ajustar las dimensiones de las nuevas placas a las del modelo previsto aunque conservando su número de matrícula.

2.2 Otras matrículas

2.2.1 Las matrículas especiales

Como hemos comentado, el Reglamento General de Vehículos, que entra en vigor el 27 de junio de 1999, modifica todos los sistemas de matrículas existentes hasta ese momento, a la vez que incorpora la matriculación de los ciclomotores, que hasta entonces era competencia de los ayuntamientos.

Los sistemas de numeración, básicamente, son comunes a todos los tipos de matrículas especiales. Se eliminan las siglas provinciales y se pasa a un sistema alfanumérico que combina cuatro números y tres letras. Las letras, como ocurre con la serie normal, elimina las vocales, la Ñ y la Q; y son las mismas 20 que en los vehículos normales.

Se diferencian entre sí por el color del fondo de la placa, por el color de los números y por una sigla que precede a la combinación alfanumérica.

Se adopta la C para ciclomotores (dígitos negros sobre fondo amarillo), E para vehículos especiales (dígitos rojos sobre fondo blanco), H para vehículos históricos (dígitos negros sobre fondo blanco), P para autorizaciones temporales para particulares (dígitos blancos sobre fondo verde), R para remolques (dígitos negros sobre fondo rojo), S para autorizaciones temporales de empresas para vehículos nuevos (dígitos blancos sobre fondo rojo), T para matrículas turísticas (dígitos negros sobre fondo blanco) y V para autorizaciones temporales de empresas para vehículos ya matriculados (dígitos blancos sobre fondo rojo). Los últimos tres tipos llevan incorporada una banda de validación que incluye el mes en números romanos y los dos últimos dígitos del año en que caduca la autorización para circular.

Los taxis y autobuses tienen otra placa además de la normal, con las letras SP (servicio público). Si el vehículo es histórico, y se ha matriculado con una placa de nuevo formato, aparece primero una letra H en la placa. Los vehículos antiguos o de colección matriculados como históricos en los que se desea respetar la matriculación antigua deben llevar además un distintivo similar, pero con la inscripción VH (vehículo histórico).



Figura 4. Distintos tipos de matrículas especiales.

2.2.2 Otras

- PME: Parque Móvil del Estado
- PMM: Parque Móvil Ministerial (en desuso)
- SH: Provincia del Sáhara Español (en desuso)
- FP: Provincia de Fernando Poo (en desuso)
- RM: Plaza de Río Muni (en desuso)

- ME: Provincia del Marruecos Español (en desuso)
- TA: Plaza de Tánger (en desuso)
- SI: Plaza de Sidi Ifni (en desuso)
- MOP: Ministerio de Obras Públicas (en desuso)
- MF: Ministerio de Fomento
- MMA: Ministerio del Medio Ambiente
- ET: Ministerio de Defensa (Ejército de Tierra)
- EA: Ministerio de Defensa (Ejército del Aire)
- FN: Ministerio de Defensa (Fuerzas Navales)
- FAE: Fuerzas Aliadas en España (Cuarteles Generales de la OTAN)
- CNP: Cuerpo Nacional de Policía (ej. CNP 0000 AU) (anteriormente DGP: Dirección General de Policía)
- CME: Cos Mossos d'Esquadra ej. CME 0000)
- PGC: Parque de la Guardia Civil (ej. PGC 0000 y dependiendo de la última letra, una subdivisión propia del cuerpo)
- Ez: Ertzaintza (Policía Autónoma Vasca) (ej. Ez 0000)
- H: Vehículo histórico (ej. H 0000 BBB)
- VH: Vehículo histórico (en desuso desde el 17 de septiembre del año 2000) (ej. VH M 0052)
- T: Turística (ej. T 0000 BBB)
- VE: Vehículo especial (ej. E 0000 BBB)

3 Reconocimien to Automático de Matrículas ANPR

El objetivo de este capítulo consiste en explicar qué es y cómo funciona un sistema ANPR, así como crear una definición del sistema que describa lo que queremos desarrollar.

3.1 Introducción

El **reconocimiento automático de matrículas** (*Automatic number plate recognition* o **ANPR** en inglés) es un método de vigilancia en masa que utiliza reconocimiento óptico de caracteres (OCR) en imágenes, para leer las matrículas de los vehículos. A los sistemas ANPR también se les conoce a veces como Reconocimiento de matrículas o **LPR** (*Licence plate recognition*) .

Debido a la masiva integración de las tecnologías dentro de la mayoría de los aspectos de la vida moderna, se produjo la demanda del procesamiento de vehículos como recurso indispensable en los sistemas actuales. Se dio la necesidad de transformar la información que pueden proporcionar los vehículos entre la realidad y los sistemas información . Esto se puede conseguir a partir del agente humano, o por un sistema especialmente inteligente, el cual sea capaz de reconocer la información de los vehículos a partir de sus matrículas en un medio real y plasmarlo en medios digitales. Debido a ello, se han desarrollado diversas técnicas de reconocimiento para diversos y los ANPR son actualmente usados en diversas aplicaciones de tráfico y seguridad.



Figura 5. Equipo de una estación de control móvil.

Los sistemas pueden escanear las matrículas con una frecuencia aproximada de una por segundo en vehículos con velocidades de hasta 160 km/h. Pueden utilizar el circuito cerrado de televisión existente o radares, o unas diseñadas específicamente para dicha tarea. Son utilizadas por las diversas fuerzas de policía y como método de recaudación electrónica de peaje en las autopistas de pago, parkings, accesos, control de fronteras, rastreo de coches robados, etc.

Los ANPR utilizan un reconocimiento óptico de caracteres (OCR) en las imágenes tomadas por las cámaras. Diversas matrículas utilizan cambios en los tamaños de las fuentes y en la posición, ya que los sistemas ANPR deben poder hacer frente a estas diferencias para ser verdaderamente eficaces. Algunos sistemas más complejos pueden distinguir variantes internacionales, aunque muchos programas se adaptan a cada país individualmente ya que el procesado y reconocimiento es mucho más óptimo y eficaz de esta manera.

3.2 Aspectos físico-técnicos

Un sistema ANPR es un equipo especial formado por un software y hardware específicos, que procesa señales gráficas de entrada como imágenes estáticas o secuencias de video, y reconocen los caracteres de una matrícula a partir de estas fuentes. La parte hardware de un sistema ANPR está formado típicamente por una cámara, una unidad de procesamiento de imágenes, un disparador u obturador, una unidad de transmisión y otra de almacenamiento.

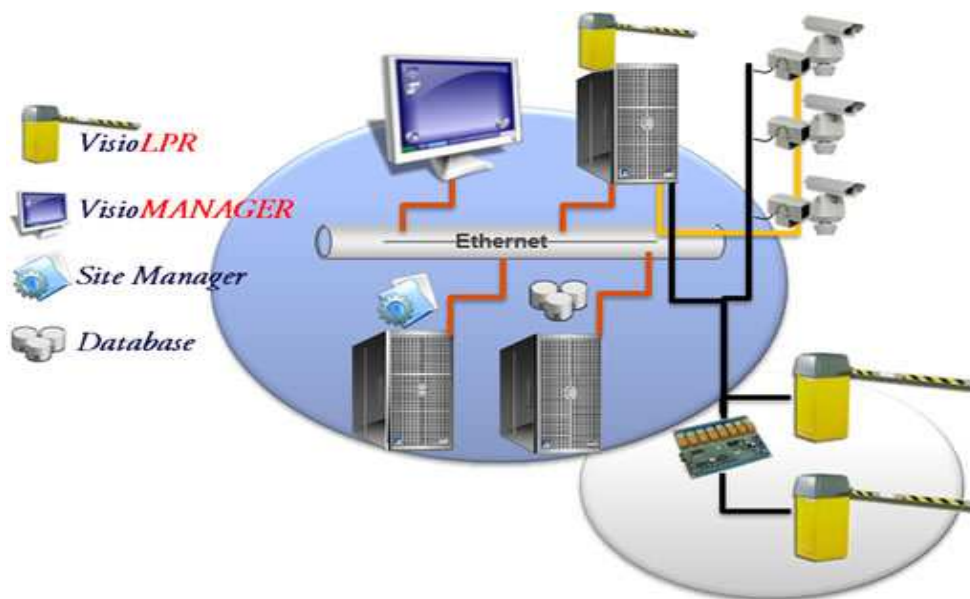


Figura 6. Diagrama de bloques de un sistema ANPR en un aparcamiento.

El disparador controla físicamente un sensor directamente instalado en un carril. Siempre que el sensor detecte un vehículo a una distancia apropiada de la cámara, éste activará el mecanismo de reconocimiento. Alternativamente a esta solución, es el software de detección o el continuo procesado de una señal video muestreada. La detección por software, o el procesado continuo de video, pueden consumir mayores recursos de sistema, pero no necesitan hardware adicional, como en el caso del hardware con disparador.

Primeramente, el sistema ANPR utiliza una serie de técnicas de manipulación de la imagen para detectar, normalizar y realzar la imagen del número de la matrícula. El procesador de imágenes reconoce instantáneas capturadas por la cámara y devuelve en un texto la representación de la matrícula detectada a través del OCR.

También se necesita asegurar la estabilidad del sistema sobre las condiciones de luz. Las cámaras corrientes no deberían ser usadas para la captura de instantáneas en la oscuridad o de noche, ya que éstas operan en un espectro visible de luz. Los sistemas ANPR a veces están basados en cámaras que operan en una banda de infrarroja del espectro. El uso de una cámara infrarroja combinada con una iluminación infrarroja es mejor para lograr este objetivo. En al menos una versión de cámara fotográfica para la supervisión de intersecciones se incluye un flash de gran alcance, que sirve para iluminar la escena y hacer que el infractor se dé cuenta de su error. Con respecto a la iluminación, las matrículas que están construidas de material reflexivo, son mucho más brillantes que el resto de las imágenes, lo que supone una mejora importante en la detección de las matrículas.

Los sistemas ANPR o ALPR se pueden utilizar de dos modos; uno permite que el proceso sea realizado en su totalidad en el lugar de la toma en tiempo real o tener sus propios procesadores de imagen dedicados (soluciones todo en uno), mientras que el otro transmite todas las imágenes de muchas cámaras a un ordenador remoto en que se realiza el proceso de OCR más tarde (ANPR genérico). Cuando se realiza *in situ*, la información capturada de la matrícula alfanumérica, fecha y hora, identificación del lugar y cualquier otra información que se requiera es completada rápidamente.

Esta información, convertida ahora en pequeños paquetes de datos, se puede transmitir fácilmente a algún ordenador remoto para un posterior procesamiento en caso de que sea necesario, o ser almacenado en el lugar para ser recuperada posteriormente. En la otra disposición, típicamente hay una gran cantidad de PC usados en una granja de servidores para manejar altas cargas de trabajo. A menudo en dichos sistemas existe la necesidad de remitir

imágenes al servidor remoto y éste puede requerir medios de transmisión con un gran ancho de banda.

Los inconvenientes de estos sistemas están centrados en el temor en cuanto a la privacidad de los movimientos de los ciudadanos y los informes de los medios sobre la identificación errónea y altas tasas de error. Sin embargo, según se han ido desarrollando, estos sistemas han logrado ser mucho más exactos y fiables.

3.3 Funcionamiento

Hay varias aplicaciones donde los ANPR pueden ser usados. Diferentes aplicaciones pueden ser bastante diferentes en términos de diseño, hardware y tecnología, e incluso para las mismas aplicaciones los fabricantes ofrecen sistemas LPR con funcionalidad similar pero con diferente estructura.

Probablemente la aplicación LPR más común son los parkings y controles de acceso. En el ámbito de las aplicaciones podemos definir un tipo común de sistema de reconocimiento de matrículas con una configuración de hardware y diseño de sistema típico.

Como ejemplo de un sistema LPR, vamos a introducir un muy simple y típico sistema para control de acceso. Como aclaración, en la mayoría de los casos, el sistema LPR es solo una parte de un sistema de control de acceso integrado.

El vehículo se acerca a la puerta del área restringida en la que dicho vehículo quiere entrar. Hay una barrera y un semáforo en rojo prohibiendo el paso. Existe un lazo inductivo instalado en la entrada para detectar la llegada, así como la presencia del vehículo. También hay una cámara CCTV o de circuito cerrado montada para vigilar la entrada.

El lazo inductivo, la cámara, el semáforo y la barrera están todos conectados a un PC de control. En este existe un software de control de acceso que coordina la operación del sistema de control.

A medida que el vehículo al lazo inductivo, este da una señal al PC de control indicándole que un vehículo ha llegado. La señal es entendida y procesada por la aplicación de control de acceso.



Figura 7. El vehículo llega a la entrada, y el lazo inductivo da una señal al PC de control.

La aplicación de control de acceso captura la señal de video de la cámara y crea una imagen digital en la memoria (RAM) del PC de control.



Figura 8. Creación de la imagen digital del vehículo.

Teniendo la imagen digitalizada, la aplicación de control de acceso avisará al modulo lector de matrículas para analizar la imagen digitalizada y leer la matrícula del vehículo. Después de su lectura, el modulo LPR devuelve la matrícula leída en código ASCII a la aplicación de control de acceso.



Figura 9. Lectura de la matrícula de la imagen digitalizada.

La aplicación de control toma la matrícula en código ASCII y la pasa a la bases de datos. El módulo de la base de datos chequea la matrícula con las diferentes listas de permisos, etc, y devuelve un marcador de “acceso permitido” o un “acceso denegado”. Dependiendo del resultado la aplicación de control abrirá la barrera y el semáforo se pondrá de color verde.

Además, la aplicación de control puede pasar datos relevantes, como la fecha y la hora exacta a la que accedió el vehículo, a la base de datos para almacenarla en un historial de acceso. Después de que el vehículo pase la puerta o no se encuentre en el lazo inductivo, el sistema volverá a estar listo para empezar el proceso entero otra vez con el siguiente vehículo que se aproxime.



Figura 10. Permiso de acceso y almacenamiento de los datos en el historial.

3.4 Aplicaciones

Existen diversas aplicaciones donde los sistemas ANPR pueden ser utilizados. Los principales valores que el reconocimiento de matrículas añade a los sistemas son la automatización y seguridad.

Después de integrar un software LPR en los sistemas inteligentes de transporte, se hace posible automatizar los peajes de autopista, analizar el tráfico, mejorar el cumplimiento de la ley, etc.

Un sistema de transporte inteligente equipado con LPR puede proporcionar:

- Sistemas de cobro de peaje flexibles y automáticos en carreteras.
- Análisis de tráfico en ciudad durante horas punta.
- Automatización de sistemas de pasaje y tráfico en movimiento.
- Mejora en la prevención de robo de vehículos.
- Aplicación efectiva de la ley y normas de circulación.
- Mayor eficiencia en los sistemas de control de fronteras.

Otras posibles aplicaciones pueden ser:

- Construcción de una base de datos completa del movimiento del tráfico.
- Automatización y simplificación de logística en aeropuertos y puertos marítimos.

- Gestión de inventario.
- Supervisión de la seguridad de las carreteras, fronteras, puntos de control, etc.
- Base de datos completa del movimiento de tráfico.
- Vigilancia de vehículos.
- Prevención de fugas o impagos en gasolineras, restaurantes drive-in, etc.

Después de integrar un software LPR en un sistema de gestión de estacionamiento, la entrada y salida de vehículos controlada y automatizada se hace posible. Por otra parte, la habilidad de reconocer el número de registro es un valor añadido significativo para las soluciones integrales de estacionamiento o la gestión de inventarios.

Un parking equipado con LPR puede proporcionar:

- Entrada y salida de vehículos flexible y automática, así como su cobro.
- Gestión de la información sobre el uso del aparcamiento.
- Mejora de la seguridad tanto para los operadores como usuarios del parking.
- Mejora del flujo de tráfico en horas punta.
- Reconocimiento del vehículo a través de la fecha y hora, así como la ubicación exacta.

3.5 Los componentes del sistema

En un sistema ANPR común, la primera tarea es la grabación de la secuencia de video, la cual no es tratada ni mostrada en el presente proyecto. Dicha secuencia de video es considerada como la entrada del sistema.

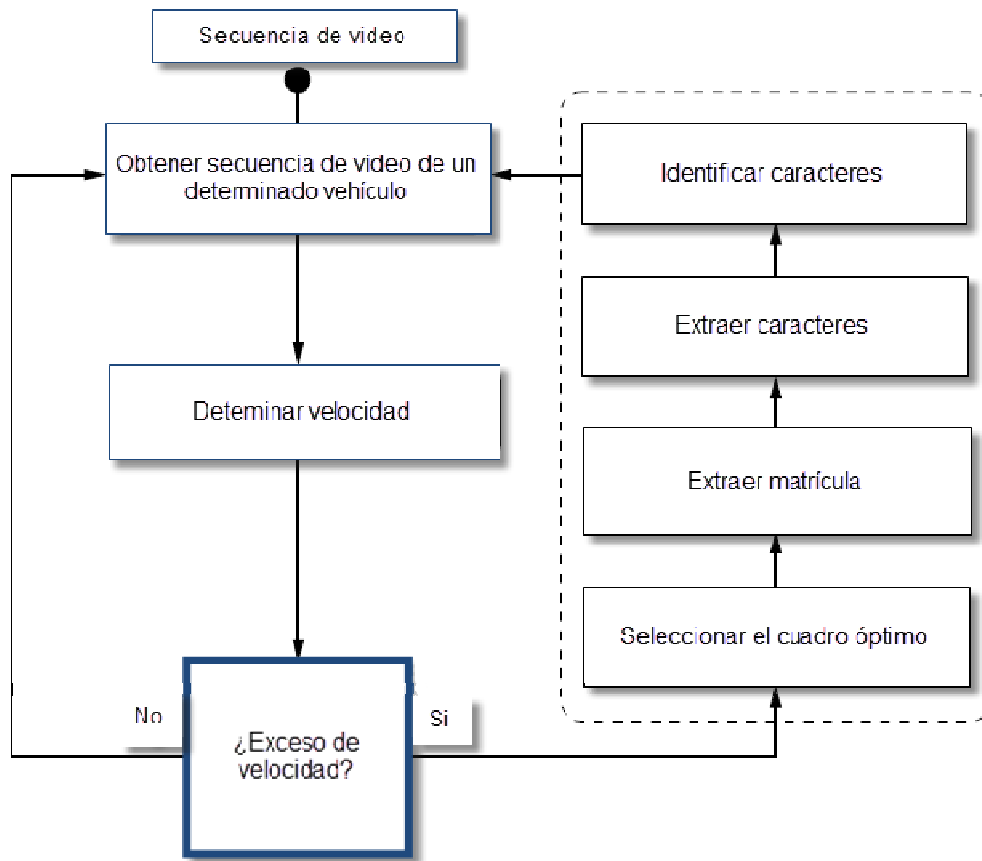


Figura 11. Procesos del sistema.

Básicamente el sistema consiste en una secuencia de 6 procesos, que idealmente son llevadas a cabo una vez por cada vehículo en una secuencia o película. La primera tarea consiste en extraer la siguiente subsecuencia de la película que contiene el siguiente vehículo. Ésta será entonces analizada y se determinará la velocidad del vehículo. En el caso donde no se exceda la velocidad permitida para la vía, ningún procedimiento adicional será necesario, y la siguiente subsecuencia se recupera para su procesamiento.

Si se excede la velocidad permitida para la vía, la matrícula del vehículo debe ser identificada. La parte del sistema encargada de este proceso se muestra enmarcada por líneas discontinuas de la Figura 11. Procesos del sistema. El subsistema consiste en cuatro tareas; la primera selecciona el cuadro óptimo de la secuencia de video entrante, la segunda extrae la región en la que se cree que contiene la matrícula, la tercera extrae o aísla los siete caracteres de la matrícula y la cuarta identifica estos mismos. Cada uno de estos procesos globales pueden estar divididos a su vez en varios subprocesos, dependiendo la complejidad del sistema ANPR. Estos subprocesos es a lo que en el presente proyecto vamos a llamar algoritmos, ya que se corresponden con las funciones de código que se han desarrollado para realizarlo

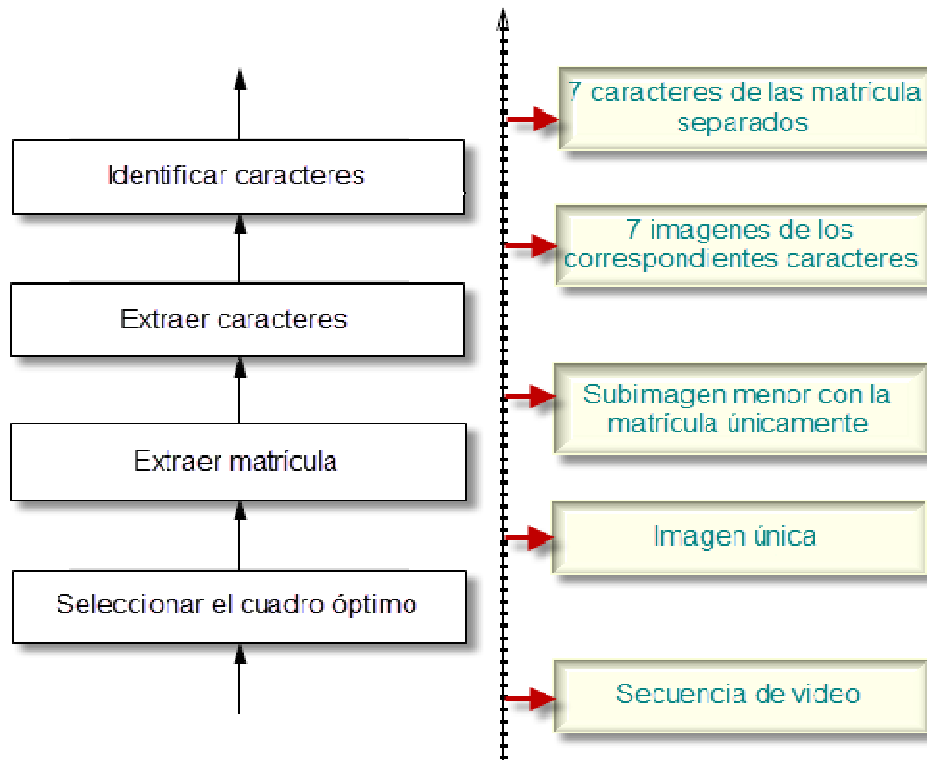


Figura 12. Nivel de extracción y procesamiento de datos.

La mitad derecha de la imagen anterior muestra la entrada y la salida de estos pasos, desde la secuencia de video de entrada hasta el final con los caracteres identificados de la matrícula. Alternativamente esta progresión puede ser vista como la reducción o la supresión de información no deseada de la señal portadora de información, lo que supone que una secuencia de video contiene grandes cantidades de información irrelevante, para extraer símbolos en la forma de caracteres de una matrícula. Este proceso de eliminación de información irrelevante, hace que el reconocimiento sea más rápido, eficaz y liviano.

3.5.1 Enfoque del proyecto

El principal objetivo de este proyecto, es investigar como el sistema actual de captura puede ser automatizado, usando una secuencia de video a partir de una cámara de video de bajo costo.

Una vez que la secuencia de video ha sido grabada, el análisis se puede dividir en dos partes principales. La primera parte es determinar la velocidad del vehículo en cuestión. Si se determina un exceso de velocidad, se lleva a cabo la segunda parte, que consiste en el reconocimiento de la matrícula. La

extracción de la misma, separar los caracteres individuales y el reconocimiento de estos, son los temas del presente proyecto.

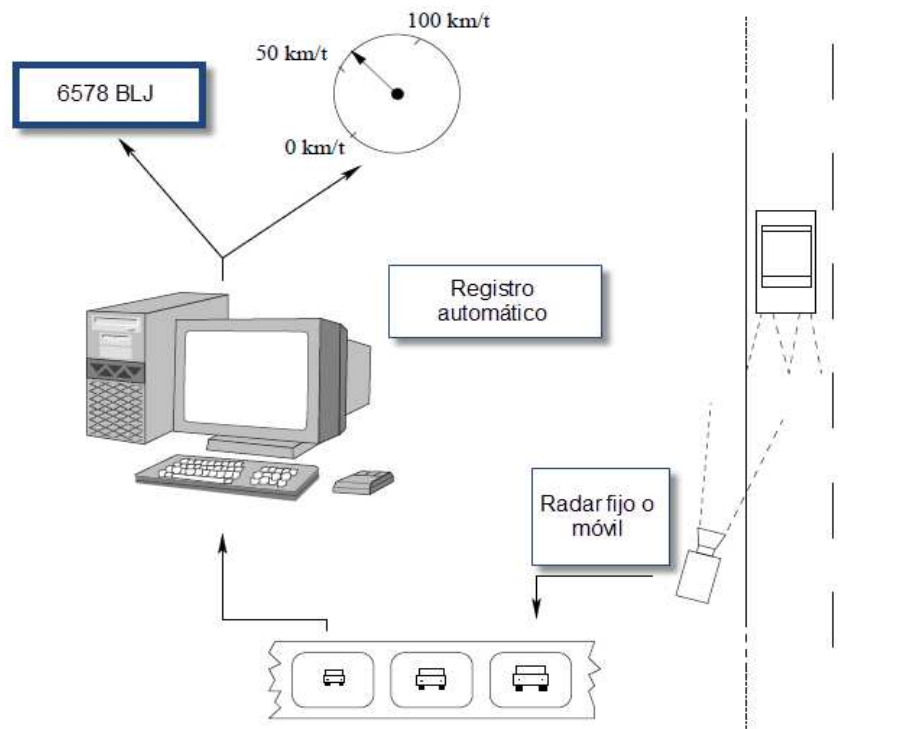


Figura 13. Flujo de trabajo del sistema.

En este proyecto nosotros mismo delimitamos los pasos antes de la extracción de la matrícula en el mejor cuadro adecuado. El asunto de qué vehículos o no cumplen con el límite de velocidad no es tratada en este proyecto. El resto del reporte solo concierne con el desarrollo de algoritmos para la extracción de la información necesaria para identificar los caracteres. La entrada es una imagen del vehículo que asumimos a un exceso de velocidad.

| Componente | Proyecto |
|--|----------|
| Velocidad | |
| - Grabación de Video | |
| - Obtención de velocidad | |
| - Transporte de datos | |
| Reconocimiento | |
| - Elección del cuadro óptimo | |
| - Extracción de la matrícula | SI |
| - Extracción de los caracteres de la placa | SI |
| - Identificación de caracteres | SI |

Tabla 1. Áreas del sistema tratadas en el proyecto.

La cuestión de cómo las secuencias de entrada se transfieren al equipo de procesado antes del análisis no se trata en el siguiente proyecto. Sin embargo es razonable pensar que los puesto de vigilancia permanentes tiene algún tipo de conexión de red fija, mientras que los temporales o móviles puedan obtener grabaciones para su posterior procesado en la central o el uso de comunicaciones móviles. Alternativamente, algún tipo de LAN inalámbrica podría ser utilizado.

3.5.2 Procesos y subprocesos del proyecto

Para el siguiente proceso se han desarrollado cinco subprocesos o algoritmos principales que el software utiliza para identificar una matrícula:

- Extracción de la matrícula:
 - **Localización de la matrícula**, encuentra y aísla el área de la matrícula del resto de la imagen.
 - **Orientación y tamaño de la matrícula**, compensa los ángulos que hacen que la matrícula parezca "torcida" y ajusta las dimensiones al tamaño requerido.
 - **Umbralización**, ajusta el brillo y el contraste de la imagen.
- Extracción de los caracteres:
 - **Segmentación de los caracteres**, encuentra los distintos caracteres presentes en la matrícula.
- Identificación de los caracteres:
 - **Reconocimiento óptico de caracteres o OCR**.
 - **Análisis sintáctico y geométrico**, comprueba los caracteres encontrados y sus posiciones con las reglas específicas del país al que pertenece la matrícula. En nuestro caso, el proyecto se centra en la matrículas españoles posteriores al año 2000, por lo que este paso está predefinido.

3.6 Aspectos matemáticos

En la mayoría de los casos, los vehículos son identificados por sus matrículas, las cuales son fácilmente reconocibles por personas, pero no por las máquinas. Para una máquina, una matrícula es solo una foto gris definida

como una función bidimensional $f(x,y)$, donde x e y son coordenadas espaciales, y f es la intensidad de brillo en un determinado punto. Debido a esto, es necesario diseñar un mecanismo matemático robusto, el cual sea capaz de extraer las semánticas del dominio espacial de la imagen capturada. Estas funciones son implementadas en los sistemas ANPR y suponen o significan la transformación de datos del dominio continuo al dominio discreto para que puedan ser interpretados por los sistemas digitales.

3.7 Dificultades

El software debe ser capaz de afrontar diferentes dificultades posibles, que incluyen:

- *Resolución de imagen pobre*, a menudo porque la matrícula está demasiado lejos, o el uso de una cámara de baja calidad.
- *Imágenes desenfocadas*, en particular desenfoque de movimiento o *motion blur*
- *Iluminación pobre y bajo contraste* debido a sobreexposición, reflexión o sombras
- Un objeto que oscurece (parte de) la matrícula, a menudo una barra del remolque, o suciedad en la matrícula
- *Técnicas de evasión*, como incrementar las propiedades de reflexión de las letras o taparlas, como las más comunes.



Figura 14. Ejemplo de matrículas conflictivas por diversos factores.

Aunque algunos de estos problemas se pueden corregir en el software, se dejan sobre todo en el lado del hardware del sistema para ofrecer soluciones a estos problemas. El aumento de la altura de la cámara puede evitar problemas con los objetos (tales como otros vehículos) que oscurecen la placa, pero introduce y aumenta otros problemas como el ajuste según la oblicuidad creciente de la placa.

Muchos países utilizan matrículas retroreflectivas. Esto devuelve la luz hacia la fuente y mejora así el contraste de la imagen. En otros los caracteres de la

matrícula no son reflectantes, dando un alto nivel del contraste con el fondo reflectante bajo cualquier condición de iluminación. Una cámara que utiliza imagen infrarroja (con un filtro normal de color sobre la lente y una fuente luminosa infrarroja al lado de ella) beneficia en gran medida, reflejándose las ondas infrarrojas desde la matrícula. Sin embargo, esto sólo es posible en cámaras ANPR dedicadas, por lo que las cámaras usadas para otros propósitos deben confiar en mayor medida en las capacidades del software. Además, cuando se necesita una imagen a todo color y la captación de detalles es necesario tener una cámara con infrarrojos y una cámara normal (en color) funcionando conjuntamente.

Debido a que uno de los principales campos de aplicación de estos sistemas es en carreteras, es necesario utilizar una cámara especial con un obturador extremadamente corto. Aparte de lo anterior, la calidad de las instantáneas capturadas serán degradadas por un no deseado efecto de desenfoque de movimiento (o en inglés y técnicamente conocido como motion blur), causado por el movimiento del propio vehículo.

Para evitar el desenfoque es ideal tener la velocidad del obturador de la cámara fijada a 1/1000 segundos. Debido a que el coche está en movimiento, el uso de velocidades más reducidas podría dar lugar a una imagen demasiado borrosa para ser leída con el software OCR, especialmente si la cámara está en una posición mucho más alta que el vehículo. Cuando el tránsito es lento o cuando la cámara fotográfica está a una altura inferior y el vehículo está en un ángulo de aproximación a la cámara, no es necesario que la velocidad del obturador sea tan alta. Velocidades del obturador de 1/500 pueden funcionar correctamente con vehículos con una velocidad de hasta 64 km/h y 1/250 hasta 8 km/h.



Figura 15. (a) Efectos de los materiales reflexivos en una matrícula y (b) Efecto de motion blur.

Algunos sistemas a escala reducida permiten algunos errores en la matrícula. Cuando se utiliza para ofrecer acceso específico de los vehículos a una zona con barrera, la decisión puede ser tomada con un índice de error aceptable de un carácter. Esto es así porque la probabilidad de que un coche desautorizado con una matrícula tan similar se considera que es absolutamente pequeña. Sin embargo, este nivel de imprecisión no sería aceptable en la mayoría de las aplicaciones de un sistema ANPR.

4 Principios de detección de áreas

El primer paso en un proceso ANPR es la detección del área de la matrícula. Este proceso incluye algoritmos que son capaces de detectar el área rectangular de una matrícula en la imagen original. Los seres humanos definen una matrícula como *"una pequeña lámina de plástico o metal adherida a un vehículo con fines identificativos"*, pero las máquinas no entienden esta definición. Debido a esto, es necesaria encontrar una definición alternativa para que una matrícula sea comprensible por las máquinas.

Definimos matrícula como *"un área rectangular de bordes horizontales y verticales"*. La alta densidad de bordes horizontales y verticales en una pequeña área es en la mayoría de los casos causado por el contraste de caracteres de una matrícula, pero no en todos. Este proceso en ocasiones puede detectar un área errónea que no se corresponde a una matrícula. A causa de ello, a veces detectamos diversas regiones candidatas para la matrícula a partir de este algoritmo, y entonces se elige la mejor de ellas a partir de un análisis heurístico adicional.

Una instantánea, que se va a comportar como un parámetro de entrada, está definida por una función $f(x,y)$, donde x e y son las coordenadas espaciales, y f es la intensidad de brillo en dicho punto. Esta función siempre va a ser discreta, ya que estamos hablando de datos digitalizados, tal que $x \in \mathbb{N} \cap y \in \mathbb{N}$, donde \mathbb{N} denota el conjunto de números naturales, incluyendo cero. Se definen operaciones como detección de bordes o rango de filtrado como transformaciones matemáticas de la función f .

La detección del área de una matrícula consiste en una serie de operaciones convolutivas y una vez modificada la instantánea se proyecta sobre los ejes x e y . Estas proyecciones son usadas para determinar el área de una matrícula.

4.1 Detección de bordes y rango de filtrado

Podemos usar una convolución periódica de la función f con tipos de matrices específicas \mathbf{m} para detectar varios tipos de borde en una imagen:

$$f'(x,y) = f(x,y) \ast \mathbf{m}[x,y] = \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} f(x,y) \cdot \mathbf{m}[\text{mod}w(x-i), \text{mod}h(y-j)]$$

donde w y h son las dimensiones de la imagen representada por la función f

Nota: La expresión $m[x,y]$ representa el elemento de la columna x y de la fila y de la matriz m .

4.1.1 Convolución de matrices

Cada operación de imagen (o filtro) está definida por una convolución de matrices. La convolución de matrices define cómo el pixel específico es afectado por los píxeles vecinos en el proceso de convolución.

Las celdas individuales en la matriz representan las vecinas relacionadas con el pixel situado en el centro de la matriz. El pixel representado por la celda y en la imagen destino, está afectado por los pixeles $x_0...x_8$ de acuerdo con la fórmula:

$$y = x_0 \times m_0 + x_1 \times m_1 + x_2 \times m_2 + x_3 \times m_3 + x_4 \times m_4 + x_5 \times m_5 + x_6 \times m_6 + x_7 \times m_7 + x_8 \times m_8$$

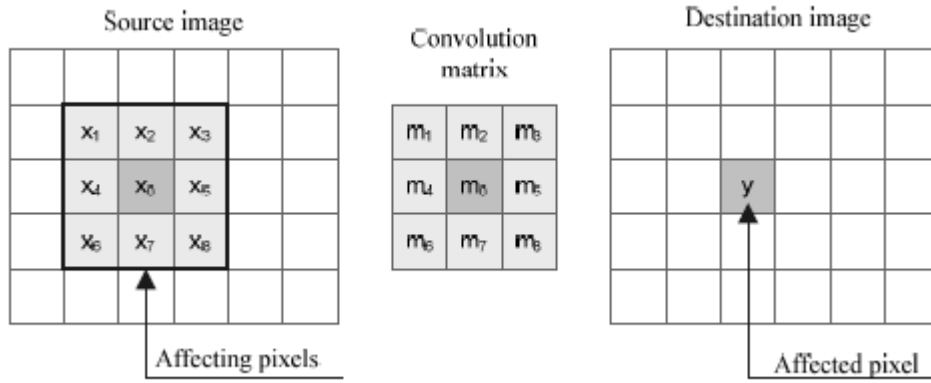


Figura 16. El píxel está afectado por sus vecinos conforme a la matriz de convolución.

A la matriz de convolución también se la denomina “kernel”.

4.1.1.1 Detección vertical y horizontal de bordes

Para detectar los bordes horizontales y verticales, convolucionamos la imagen original con las matrices m_{he} y m_{ve} . El resultado de la convolución de matrices es usualmente mucho más pequeño que la imagen original. También, podemos usar matrices más grandes para detectar bordes más abruptos.

$$m_{he} = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}; m_{ve} = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

4.1.1.2 Algoritmo de detección de bordes Sobel

El operador Sobel es el detector de bordes por antonomasia y es el ejemplo básico de los algoritmos de detección de bordes. Usa un par de matrices de 3x3 para la convolución de matrices. La primera de ellas evalúa bordes verticales y la segunda bordes horizontales.

$$G_x = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}; G_y = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

El módulo del pixel afectado se calcula a partir de la fórmula $|G| = \sqrt{G_x^2 + G_y^2}$. En la práctica es más rápido calcular solo una magnitud aproximada como $|G| = |G_x| + |G_y|$.



Figura 17. Filtrado Sobel horizontal y vertical.

4.1.1.3 Algoritmo de detección de bordes Canny

El algoritmo de Canny es generalmente conocido como el mejor y más completo método de detección de bordes desarrollado hasta la fecha. Canny tuvo como objetivo un detector de bordes que satisficiera tres criterios clave:

- *Una baja tasa de errores.* En otras palabras, es importante que los bordes en las imágenes no se pierdan y que no deba haber ninguna respuesta donde no existan.
- *Los puntos de borde detectados deberían estar bien localizados.* En otras palabras, la distancia entre los píxeles del borde encontrados por el detector y el borde real, debe ser mínima.
- *Debería haber solo una respuesta para un solo borde.*

Sin entrar de fondo en el análisis matemático detallado y el razonamiento realizado por Canny, el procedimiento básico se puede resumir en los siguientes pasos:

1. *La imagen es primeramente suavizada usando un filtro kernel gaussiano:* Los operadores gradiente son sensibles al ruido y este paso preliminar se realiza para reducir el ruido de la imagen. Cuanto mayor es el ancho del kernel, mayor es el suavizado o reducción de ruido que se logra. Sin embargo, grandes kernels provocan un error mayor en la localización de los bordes.

2. *Encontrar el grosor del borde:* Esto se consigue obteniendo el gradiente de la imagen con el operador Sobel en la dirección horizontal y vertical, y añadiendo después la magnitud de estos componentes como una medida del 'grosor del borde', como $E(x, y) = |G_x(x, y)| + |G_y(x, y)|$

3. *Calcular la dirección del borde:* Se calcula fácilmente como:

$$\theta = \tan \frac{G_y(x, y)}{G_x(x, y)}$$

4. *Digitalizar la dirección del borde:* Una vez que se conoce la dirección, la aproximamos a una dirección de borde que puede ser trazada en una imagen digital. Considerando un pixel arbitrario, la dirección de un eje a través de este pixel puede tomar uno de los cuatro valores posibles – 0°(vecinas de este a oeste), 90°(vecinos de norte a sur), 45°(vecinos de noreste a suroeste), 135°(vecinos de noroeste a sureste). Por consiguiente, aproximamos el ángulo θ calculado por cualquiera de estos cuatro ángulos cercanos a este valor. En algunas formulaciones, para este paso, la orientación del borde se calcula a partir de la precisión del subpixel usando una conectividad de 8 píxeles.

5. *Supresión no máxima:* Después de que las direcciones son conocidas, la *supresión no máxima* se aplica. Esto funciona trazando a lo largo de la dirección del borde y eliminando algún valor del pixel (es decir, ponerlo a 0) que no se considere borde. Esto representará una fina línea en la imagen de salida.

6. *Histéresis:* Después de los primeros cinco pasos han sido completados, el paso final es realizar el seguimiento a lo largo de los píxeles restantes que no han sido suprimidos y el umbral de la imagen para identificar los píxeles del borde. Sin embargo, lo que es crítico para el método Canny, es el uso de dos umbrales distintos, un

valor más alto T_2 y otro más bajo T_1 . El destino de cada pixel se determina de acuerdo con los siguientes criterios:

- Si $|E(x,y)| < T_1$, el pixel es rechazado y no es un pixel del borde;
- Si $|E(x,y)| > T_2$, el pixel es aceptado y es un pixel del borde;
- Si $T_1 < |E(x,y)| < T_2$, el pixel es rechazado salvo que la trayectoria consistente de pixeles de un borde se conecta con un pixel de borde incondicional $|E(x,y)| > T_2$.

Dos comentarios son pertinentes a la aplicación práctica del detector Canny. Primero, nos encontramos, en general, interesados en la identificación de bordes en una particular escala. Podemos introducir la idea de escala característica con un kernel de filtrado gaussiano de varios anchos. Grandes kernels introducen una gran cantidad de suavizado de imagen y los gradientes en consecuencia son reducidos. Es normal, por tanto, especificar el tamaño de suavizado kernel en el algoritmo de Canny para reflejar este objetivo. En segundo lugar, se observa que el detector Canny identifica bordes débiles ($T_1 < |E(x,y)| < T_2$) solo si están conectados a bordes gruesos ($|E(x,y)| > T_2$).

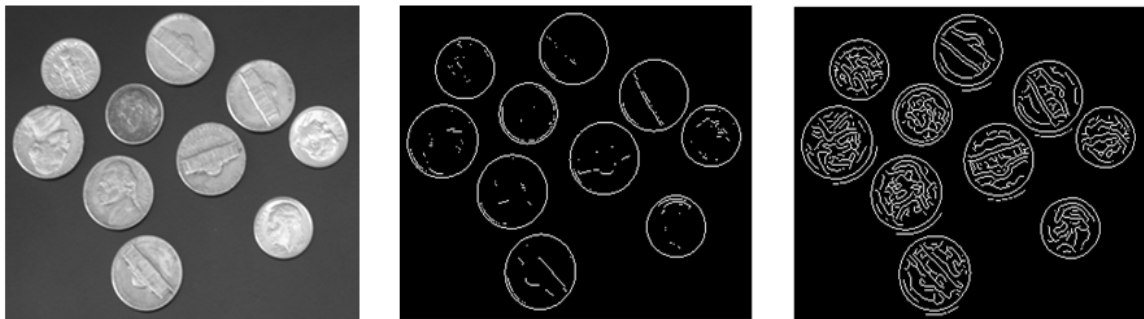


Figura 18. Filtrados Sobel y Canny respectivamente.

4.1.1.4 Filtrado de promedio/rango horizontal y vertical

Los filtros con orientaciones verticales y horizontales son a veces usados para detectar grupos de bordes con alta densidad de brillo en la matrícula. La anchura del filtro orientado horizontalmente es mucho más grande que la anchura de la matriz ($w \gg h$), y viceversa para el filtro vertical ($w \ll h$).

Para preservar la intensidad global de una imagen, es necesario que cada pixel sea reemplazado con la media de la intensidad del pixel en el área

cubierta por el filtro. En general, la matriz de convolución debe cumplir la siguiente condición:

$$\sum_{i=0}^{w-1} \sum_{j=0}^{h-1} m_{hr}[i, j] = 1.0$$

Donde w y h son dimensiones de la matriz.



Figura 19. Varios tipos de filtrados promedios y de detección de bordes direccionales.

4.2 Proyección horizontal y vertical de la imagen

Después de las operaciones de convolución, podemos detectar un área de la matrícula acorde a las estadísticas de la instantánea. Hay varios métodos de análisis estadísticos. Uno de ellos es una proyección horizontal y vertical de una imagen en los ejes x e y .

La proyección vertical de una imagen es un gráfico, que representa una magnitud global de la imagen conforme al eje y . Si calculamos la proyección vertical de la imagen después de aplicar un filtro de detección de bordes también vertical, la magnitud de cierto punto representa el hecho de los ejes verticales en ese punto. Por tanto, la proyección vertical de la imagen transformada puede ser usada para una localización vertical de la matrícula. La

proyección horizontal representa una magnitud global de la imagen trazada en el eje x de coordenadas.

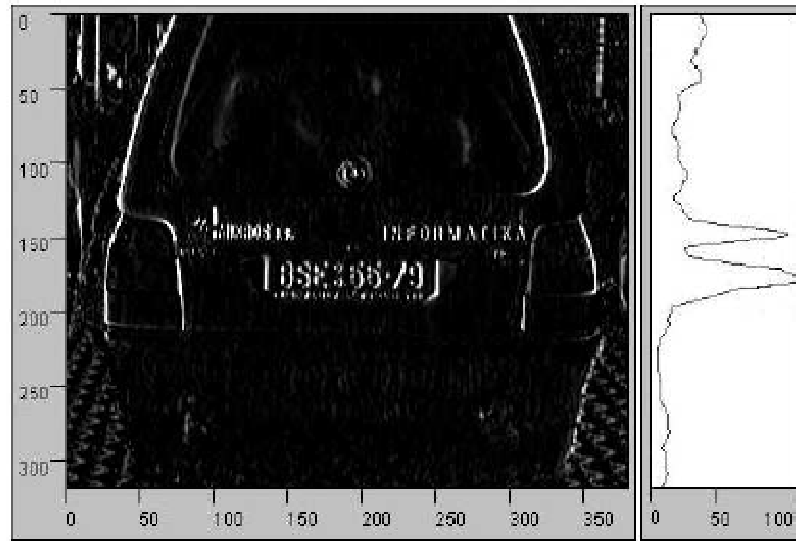


Figura 20. Proyección vertical de una imagen en el eje y.

Sea una imagen de entrada definida por una función discreta $f(x,y)$. Entonces, una proyección vertical p_y de una función f en un punto y , es un conjunto de los valores de la correspondiente fila y de la imagen de entrada. Similarmente, una proyección horizontal en un punto x de esa función es un conjunto de todos los valores en la columna x .

Las proyecciones verticales y horizontales se pueden definir matemáticamente como:

$$p_x(x) = \sum_{j=0}^{h-1} f(x,j); \quad p_y(y) = \sum_{i=0}^{w-1} f(i,y)$$

Donde w y h son las dimensiones de la imagen.

4.3 Análisis estadístico de doble fase

El análisis estadístico está compuesto por dos fases. La primera fase cubre la detección de una gran área de la matrícula. Este área es entonces enderezada y procesada en la segunda parte del análisis. La salida de este análisis es el área exacta de la matrícula. Estas dos fases están basadas en el

mismo principio, pero hay diferencias en los coeficientes que son usados para determinar los límites de área recortadas.

La detección del área de la matrícula consiste en un “recorte de banda” y “recorte de placa”. El recorte de banda es una operación usada para detectar y recortar el área de la matrícula (también llamada banda) por el análisis de la proyección vertical de la instantánea. El recorte de la placa es la operación consecuente, que es usada para detectar y recortar la placa de la banda/franja (no de la instantánea entera) a través de un análisis horizontal de dicha banda.

4.3.1 La Instantánea

Suponemos que la instantánea está representada por la función $f(x, y)$, donde $x_0 \leq x \leq x_1$ y $y_0 \leq y \leq y_1$. El $[x_0, y_0]$ representa la esquina superior izquierda de la imagen, y $[x_1, y_1]$ representa la esquina inferior derecha. Si w y h son las dimensiones de la imagen, por lo que $x_0=0, y_0=0, x_1=w-1$ y $y_1=h-1$.

4.3.2 La franja

La franja b en la instantánea es un rectángulo arbitrario $b=(x_{b0}, y_{b0}, x_{b1}, y_{b1})$, tal que:

$$(x_{b0} = x_{\min}) \cap (x_{b1} = x_{\max}) \cap (y_{\min} \leq y_{b0} < y_{b1} \cap y_{\max})$$

4.3.3 La placa

De manera similar, la placa p en la franja b es un rectángulo arbitrario $p = (x_{p0}, y_{p0}, x_{p1}, y_{p1})$, tal que:

$$(x_{b0} \leq x_{p0} \leq x_{p1} \leq x_{b1}) \cap (y_{p0} = y_{b0}) \cap (y_{p1} = y_{b1})$$

La franja también puede ser definida como una selección vertical de la instantánea y la placa como una selección horizontal de la franja. La figura 21 Demuestra esquemáticamente este concepto:

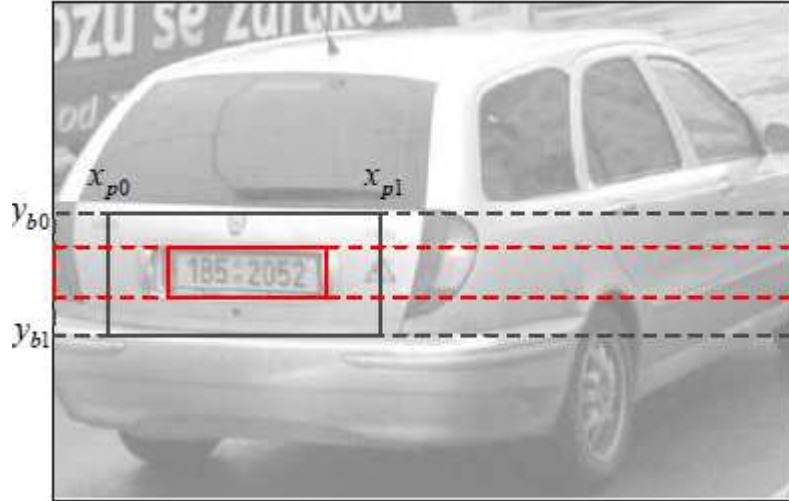


Figura 21. Recorte de placa de dos fases.

Como podemos ver en la Figura 21. Recorte de placa de dos fases, el color negro representa la primera fase de recorte y el color rojo la segunda. Las franjas están representadas por líneas discontinuas y las “placas” por continuas.

4.3.4 Detección vertical – recorte de franja

La primera y segunda fase del recorte están basadas en el mismo principio. El recorte de franja es una selección vertical de la instantánea de acuerdo al análisis de la gráfica de la proyección vertical. Si h es la altura de una imagen analizada, su correspondiente proyección vertical $p_y^r(y)$ contiene h valores tales que $y \in \langle 0, h - 1 \rangle$.

El gráfico de la proyección puede ser a veces bastante imperfecto para el análisis, debido a una gran dispersión estadística de los valores $p_y^r(y)$. Hay dos enfoques diferentes para resolver este problema. Podemos desenfocar la imagen de entrada (solución costosa), o podemos disminuir la dispersión estadística de la proyección irregular p_y^r convolucionando dicha proyección con un vector promedio:

$$p_y(y) = p_y^r(y) \ast \mathbf{m}_{hr}[y]$$

donde \mathbf{m}_{hr} es el vector promedio, análogo al promedio horizontal explicado en el apartado 4.1.1.4. Generalmente, la anchura del vector es nueve en la configuración por defecto.

Después de la convolución con el vector promedio, la proyección vertical de la imagen puede ser algo similar a lo siguiente:

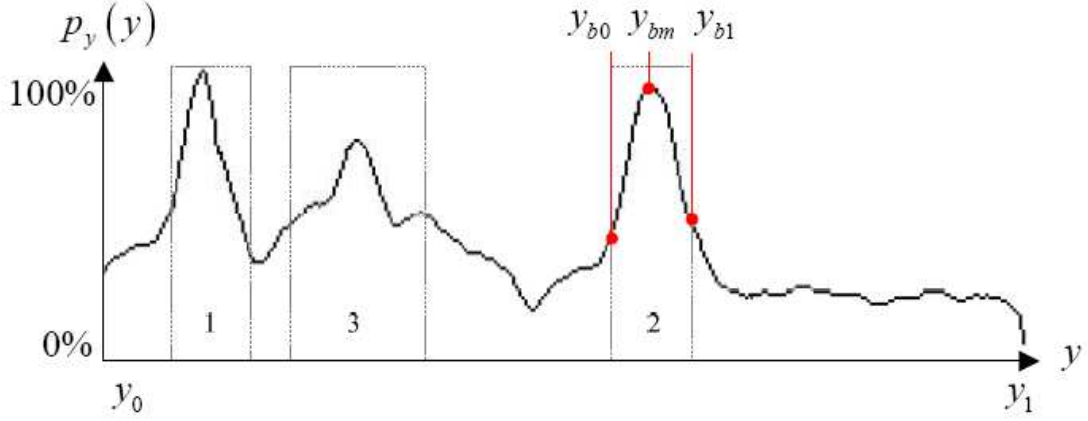


Figura 22. Proyección vertical de la imagen convolucionada con una matriz de promedio.

El problema fundamental del análisis es calcular los picos en la gráfica de la proyección vertical. Los picos se corresponden a las bandas o franjas, que son las posibles matrículas candidatas. El valor máximo de $p_y(y)$ correspondiente a un eje de la banda, puede ser calculado como:

$$y_{bm} = \arg \max_{y_0 \leq y \leq y_1} [p_y(y)]$$

y_{b0} y y_{b1} son coordenadas de la franja, la cual puede ser detectada como:

$$y_{b0} = \max_{y_0 \leq y \leq y_{bm}} [y | p_y(y) \leq c_y \cdot p_y(y_{bm})]$$

$$y_{b1} = \min_{y_{bm} \leq y \leq y_1} [y | p_y(y) \leq c_y \cdot p_y(y_{bm})]$$

c_y es una constante, que se usa para determinar la base del pico y_{bm} . En la práctica la constante es calibrada a $c_1=0.55$ para la primera fase de detección, y $c_2=0.42$ para la segunda.



Figura 23. Franja detectada por el análisis de la proyección vertical.

Este principio se aplica de manera iterativa para detectar varias bandas posibles. y_{b0} y y_{b1} son calculadas en cada paso del proceso iterativo. El recorte de franja está seguido de una operación que detecta placas en esa misma franja.

4.3.5 Detección horizontal – recorte de placa

A diferencia del recorte de franja o banda, existen algunas diferencias entre la primera y segunda fase del recorte de la placa.

4.3.5.1 Primera fase

Existe una fuerte analogía entre el recorte de banda y placa. El recorte de placa está basado en una proyección horizontal de la franja. Primeramente, la franja o banda debe ser procesada por un filtro de detección vertical. Si w es la anchura de la banda (o la anchura de la imagen analizada), la correspondiente proyección horizontal $p_x^r(x)$ contiene w valores:

$$p_x(x) = \sum_{j=y_{b0}}^{y_{b1}} f(x, j)$$

Se debe tener en cuenta que $p_x(x)$ es una proyección de la banda, no de la imagen total. Esto puede ser conseguido por una suma en el intervalo $\langle y_{b0}, y_{b1} \rangle$, que representa los bordes verticales de la banda. Desde la proyección horizontal $p_x^r(x)$ puede haber una dispersión estadística grande, que podemos disminuir, convolucionando con un vector de promedio, de la misma manera que describimos anteriormente. La anchura del vector promedio es usualmente igual a la mitad de la anchura estimada de la matrícula.

Por lo tanto, el valor máximo que corresponde a la placa puede ser calculada como:

$$x_{bm} = \arg \max_{x_0 \leq x \leq x_1} [p_x(x)]$$

x_{b0} y x_{b1} son coordenadas de la franja, la cual puede ser detectada como:

$$x_{b0} = \max_{x_0 \leq x \leq x_{bm}} [x \mid p_x(x) \leq c_x \cdot p_x(x_{bm})]$$

$$x_{b1} = \min_{x_{bm} \leq x \leq x_1} [x \mid p_x(x) \leq c_x \cdot p_x(x_{bm})]$$

donde c_x es una constante que se usa para determinar la base del pico x_{bm} . La constante esta calibrada a $c_x=0.86$ para la primera fase de la detección.

4.3.5.2 Segunda fase

En la segunda fase, la posición horizontal de la matrícula se detecta de forma distinta. Debido a la corrección de rotado entre la primera y segunda fase del análisis, la placa de área mayor debe ser duplicada en un nuevo mapa de bits, que será lo mismo que una imagen monocroma para todos los casos. Tenemos una función $f_n(x, y)$ correspondiente a dicho mapa de bits. Esta imagen tiene un nuevo sistema de coordenadas, tal que $[0,0]$ representa la esquina superior izquierda y $[w-1, h-1]$ la esquina inferior derecha, donde w y h son sus dimensiones.

En contraste con la primera fase de detección, la placa imagen no ha sido procesada por el filtro de detección vertical. Si asumimos que la placa es blanca con los bordes negros, podemos detectarlos como transiciones de blanco a negro y viceversa. La proyección horizontal $p_x(x)$ de la imagen está ilustrada en la figura Figura 24. Proyección horizontal $p_x(x)$ y su derivada $p'_x(x)$. Para detectar las transiciones de blanco a negro o viceversa, es necesario calcular una derivada $p'_x(x)$ de la proyección $p_x(x)$. Ya que la proyección no es continua, el paso de la derivación no puede ser un número infinitamente pequeño ($h \neq \lim_{x \rightarrow 0} x$). Si derivamos una función discreta, el paso de derivación h debe ser un número entero (por ejemplo $h=4$). La derivada de $p_x(x)$ se define como:

$$p'_x(x) = \frac{p_x(x) - p_x(x-h)}{h}$$

donde $h=4$.

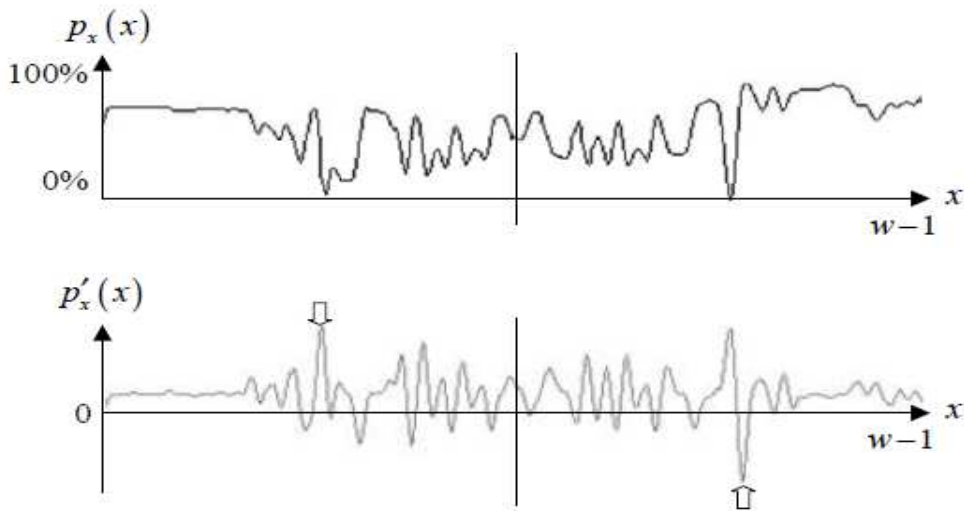


Figura 24. Proyección horizontal $p_x(x)$ y su derivada $p'_x(x)$.



Figura 25. Área mayor de la matrícula detectada después del proceso de rotación.

El borde izquierdo y derecho de la placa se pueden determinar por un análisis de la proyección $p'_x(x)$. La esquina izquierda x_{p0} está representada por una transición de negro a blanco (pico positivo en la Figura 24. Proyección horizontal $p_x(x)$ y su derivada $p'_x(x)$), y la esquina derecha x_{p1} por la transición blanco-negro (pico negativo en la Figura 24. Proyección horizontal $p_x(x)$ y su derivada $p'_x(x)$):

$$x_{p0} = \min_{0 \leq x \leq \frac{w}{2}} \{x \mid p'_x(x) \geq c_d \cdot \max_{0 \leq x < w} \{p'_x(x)\}\}$$

$$x_{p1} = \max_{\frac{w}{2} \leq x \leq w} \{x \mid p'_x(x) \leq c_d \cdot \min_{0 \leq x < w} \{p'_x(x)\}\}$$

donde c_d es una constante usada para determinar el mayor pico izquierdo negativo y el mayor pico derecho positivo. Las esquinas izquierda y derecha pueden encontrarse en las mitades opuestas de la placa detectada de acuerdo con las limitaciones $0 \leq x < \frac{w}{2}$ para x_{p0} , y $\frac{w}{2} \leq x < w$ para x_{p1} .

En esta fase del proceso de reconocimiento, no es posible elegir el mejor candidato. Esto se puede conseguir a través de un análisis heurístico de caracteres después de la segmentación.

4.4 Análisis heurístico y prioridad de selección de candidatos

En general, la imagen capturada puede contener varios candidatos de ser matrícula. Por esto, el algoritmo de detección siempre obtiene varias franjas, y varias placas de cada banda. Hay un número máximo predefinido de candidatos, que son detectados por un análisis de proyecciones.

Hay varias fórmulas heurísticas, que son usadas para determinar el costo de los candidatos seleccionados de acuerdo con sus propiedades. Estas heurísticas han sido elegidas *ad hoc* (expresamente para este problema) durante los experimentos prácticos. El reconocimiento lógico clasifica los candidatos de acuerdo a su coste, del más adecuado al menos adecuado, por lo que el

candidato más adecuado es examinado por un análisis heurístico más profundo. Este análisis acepta o rechaza definitivamente dicho candidato. Como hay una necesidad de analizar caracteres de forma individual, este tipo de análisis consume una gran cantidad de tiempo de procesado.


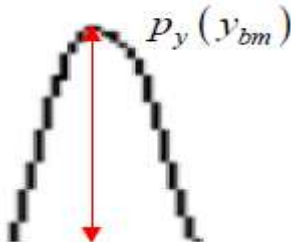
El concepto básico del análisis puede ser ilustrado en los siguientes pasos:

1. Detectar un número posible de candidatos.
2. Colocarlos de acuerdo con su coste (determinado por heurísticas básicas).
3. Cortar el primer candidato de la lista con el mejor coste.
4. Segmentarlo y analizarlo por una análisis más profundo (tiempo de procesado).
5. Si el análisis rechaza el candidato, volver al paso 3.

4.4.1 Selección de prioridades y análisis heurístico de bandas

El análisis es usado para evaluar el costo de los candidatos, y ordenarlos de acuerdo a éste criterio. Existen varias heurísticas independientes, que pueden ser usadas para evaluar el costo α_i . Las heurísticas pueden ser usadas independientemente o por separado, para calcular el coste global del candidato, a partir de la siguiente suma ponderada:

$$\alpha = 0.15 \cdot \alpha_1 + 0.25 \cdot \alpha_2 + 0.4 \cdot \alpha_3 + 0.4 \cdot \alpha_4$$

| Heurísticas | Ilustración | Descripción |
|------------------------------------|---|---|
| $\alpha_1 = y_{b0} - y_{b1} $ |  | Altura de la banda en píxeles. Las bandas con una baja altura serán prioritarias. |
| $\alpha_2 = \frac{1}{p_y(y_{bm})}$ |  | $p_y(y_{bm})$ es un valor máximo de pico en la proyección vertical de la imagen, que se corresponde con la banda procesada. Las bandas con una alta cantidad de bordes verticales serán las prioritarias. |

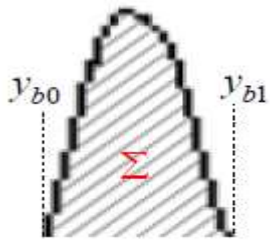

| | | |
|---|---|---|
| $\alpha_3 = \frac{1}{\sum_{y=y_{b0}}^{y_{b1}} p_y(y)}$ |  | <p>Esta heurística es similar a la anterior, pero ésta no solo tiene en cuenta el valor pico más alto, sino también el valor del <i>área</i> bajo la gráfica entre los puntos y_{b0} y y_{b1}. Estos puntos definen la posición vertical de la banda evaluada.</p> |
| $\alpha_4 = \left \frac{ x_{p0} - x_{p1} }{ x_{b0} - x_{b1} } - c \right $ |  | <p>Las proporciones de una fila de las matrículas es similar en la mayoría de los países. Si asumimos que la relación altura/anchura de la placa es alrededor de una constante $c=5$, podemos comparar la relación medida con la estimada para evaluar el costo de la matrícula.</p> |

Tabla 2. Heurísticas utilizadas en el proceso de selección de candidatos.

4.4.2 Análisis heurístico profundo

El análisis profundo determina la validez de un candidato para la matrícula. Los candidatos deben ser segmentados en caracteres individuales para extraer características sustanciales. La lista de candidatos es iterativamente procesada hasta que el primer número de la matrícula es encontrado. El candidato es considerado como una matrícula válida, si éste cumple con los requisitos para su validez.

Si tenemos en cuenta que la placa p es segmentada en varios caracteres $p_0 \dots p_{n-1}$, donde n es el número de caracteres. La variable w_i es el ancho del carácter i . Ya que todos los caracteres segmentados tienen un ancho uniforme abrupto, se puede usar la desviación estándar de esos valores como una fórmula heurística:

$$\beta_1 = \sqrt{\frac{1}{n} \sum_{i=0}^{n-1} (w_i - \bar{w})^2}$$

donde \bar{w} es una media aritmética de los anchos de los caracteres:

$$\bar{w} = \frac{1}{n} \sum_{i=0}^{n-1} w_i$$

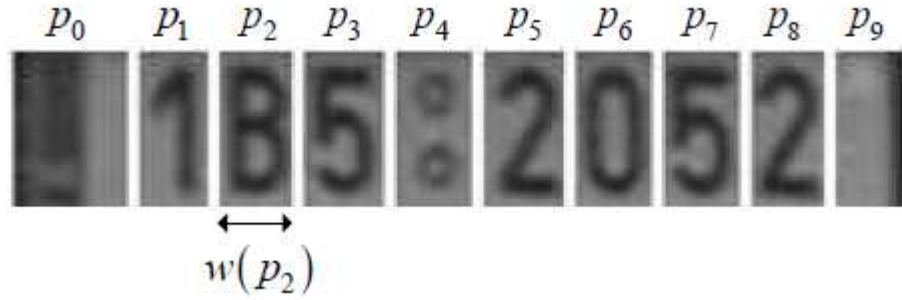


Figura 26. Segmentación en caracteres individuales de la matrícula para un análisis heurístico más profundo.

Si tenemos en cuenta que la matrícula consiste en caracteres oscuros sobre un fondo luminoso, podemos usar un histograma de brillo para determinar si el candidato satisface esta condición. Debido a que algunas placas específicas de países son oscuras o negativas, podemos usar el histograma para hacer frente a este tipo de matrículas.

Se tiene un histograma de brillos $H(b)$, donde b es un cierto valor de brillo. Las variables b_{min} y b_{max} son los valores del punto más oscuro y más brillante respectivamente. La placa será negativa cuando la heurística β_2 sea negativa:

$$\beta_2 = \left(\sum_{b=b_{mid}}^{b_{max}} H(b) \right) - \left(\sum_{b=b_{min}}^{b_{mid}} H(b) \right)$$

donde b_{mid} es un punto medio en el histograma, tal que $b_{mid} = \frac{b_{max} - b_{min}}{2}$.

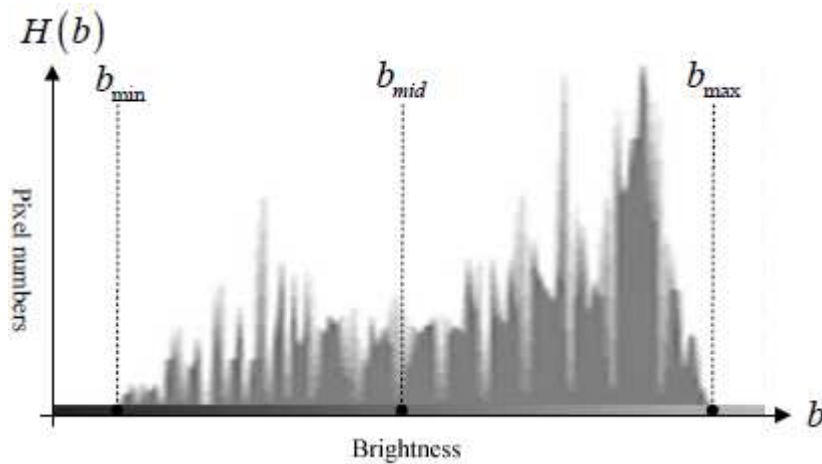


Figura 27. Histograma de brillo de la matrícula.

4.5 Mecanismo de rotación

La placa rectangular capturada puede estar rotada o distorsionada, debido al posicionamiento del vehículo hacia la cámara. Ya que este fenómeno degrada significativamente las habilidades de reconocimiento, es importante implementar mecanismos adicionales capaces de detectarlo y corregirlo.

El problema fundamental de este proceso es determinar un ángulo, bajo el cual la placa sea rotada y rectificada. Por tanto, la corrección de la placa evaluada, puede ser realizada por una transformación trivial afín.

Es importante entender la diferencia entre “distorsionado o desviado” y “girado”, ya que su tratamiento será distinto. La matrícula es un objeto tridimensional, la cual es proyectada sobre una imagen bidimensional durante la captura. El posicionamiento del objeto puede a veces causar la desviación de ángulos y proporciones.

Si la línea vertical de la placa v_p no es idéntica a la línea vertical del objetivo de la cámara v_c , la placa puede estar desviada o distorsionada. Si la líneas verticales v_p y v_c son idénticas, pero el eje a_p de la placa no es paralelo al eje de la cámara a_c , la placa estará girada o rotada.

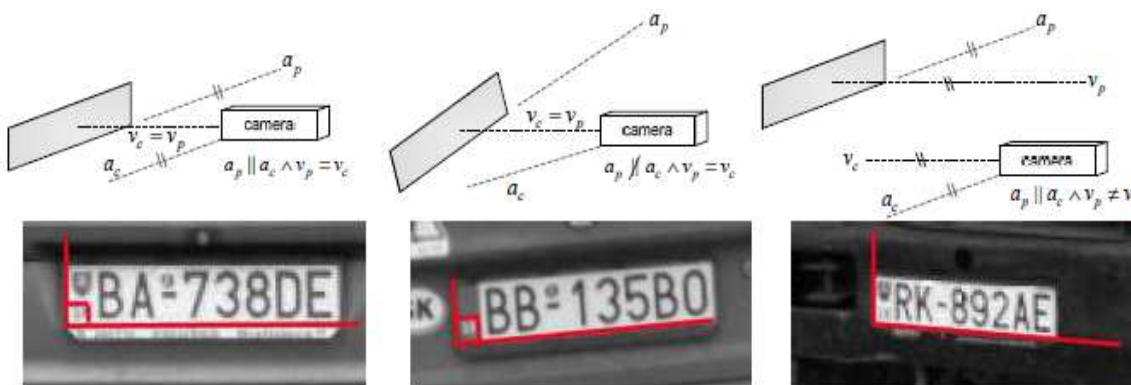


Figura 28. (a) Matrícula capturada sobre un ángulo correcto (b) Placa girada o rotada
(c) Placa distorsionada o torcida.

4.5.1 Detección de la desviación

La transformada de Hough es un algoritmo especial, que es usada para extraer una forma determinada y específica dentro de una imagen, y también para la detección de líneas. Ésta es ampliamente utilizada para diversos fines en la problemática de la visión máquina, pero en el presente proyecto se ha

usado para detectar la desviación de la placa capturada, así como para calcular su ángulo de rotación.

Es importante saber, que la transformada Hough no distingue entre conceptos como “rotación” y “desviación”. La transformada de Hough puede ser utilizada solamente para calcular un ángulo aproximado de una imagen en un dominio bidimensional.

La representación matemática de una línea en el sistema de coordenadas ortogonal es una ecuación $y = a \cdot x + b$, donde a es la inclinación y b la sección del eje x de la línea definida. Por tanto, la línea es el conjunto de todos los puntos $[x, y]$, para los cuales esta ecuación es válida. Se sabe que la línea contiene un número infinito de puntos, así como hay un número infinito de diferentes líneas, que se pueden cruzar en un cierto punto. La relación entre estas dos afirmaciones es la idea básica de la transformada Hough.

La ecuación $y = a \cdot x + b$ puede ser también escrita como $y = -x \cdot a + y$, donde x e y son parámetros. Por lo tanto, la ecuación define un conjunto de todas las líneas (a, b) , que pueden cruzarse con un determinado punto $[x, y]$. Para cada punto en el eje de coordenadas “XY”, hay una línea en un sistema de coordenadas “AB” (llamado espacio Hough).

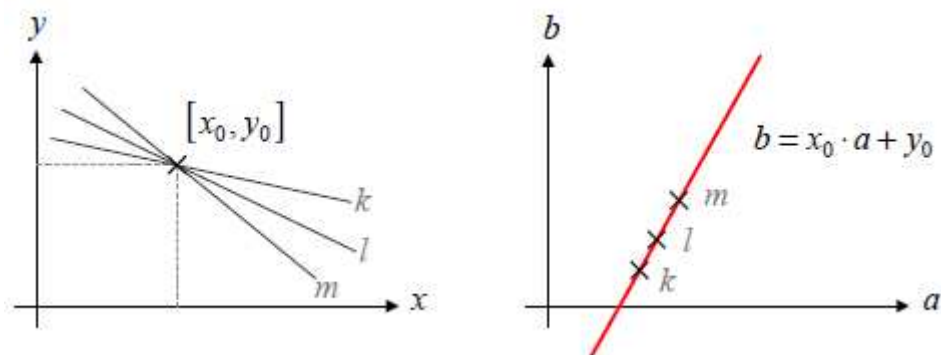


Figura 29. Sistemas de coordenadas “XY” y “AB”.

Hay varios puntos marcados como k , l y m , en el espacio Hough, que corresponden a líneas en el sistema de coordenadas “XY”, las cuales se pueden cortar con el punto $[x_0, y_0]$.

Sea una función $f(x, y)$ continua y para cada punto $[a, b]$ en el espacio Hough, hay una línea en el sistemas de coordenadas “XY”. Podemos calcular la magnitud de un punto $[a, b]$ como el recuento de todos los puntos que existen en el espacio “XY”, que se sitúan en la línea $a \cdot x + b$.

Asumimos que $f(x, y)$ es una función discreta, que representa una instantánea de dimensiones w y h . Para calcular la transformada de Hough de

una función como ésta, es necesario formalizarla en un sistema de coordenadas unificado a partir del siguiente modo:

$$x' = \frac{2 \cdot x}{w} - 1; y' = \frac{2 \cdot y}{h} - 1$$

Aunque el espacio definido por el sistema de coordenadas unificado es siempre discreto (punto flotante) en los sistemas digitales, nosotros asumiremos que es continuo. Generalmente, se puede definir la transformada de Hough $h'(a', b')$ de una función continua $f'(x', y')$ en un sistema de coordenadas unificado como :

$$h'(a', b') = \int_{-1}^1 f'(x', a' \cdot x' + b') dx'$$

Se usa la transformada de Hough de cierta imagen para evaluar su ángulo de desviación. Se puede apreciar la transformada coloreada en la Figura 30. Proceso gráfico de la transformada de Hough.

En la figura anterior, podemos observar el proceso gráfico al aplicar la Transformada de Hough. Primeramente observamos la matrícula en el sistema de coordenadas unificado después aplicar un filtro de detección de borde horizontal. Las dos representaciones siguientes muestran la Transformada de Hough de la matrícula en el sistema de coordenadas “ θB ” y la transformada coloreada en el sistema de coordenadas “ AB ” respectivamente.. Los pixeles con un relativo alto valor están remarcados con color rojo en el sistema de coordenadas “ AB ”. Cada pixel se corresponde con una larga línea blanca de una matrícula sin adaptar. Teniendo en cuenta que el ángulo de tales líneas determina el ángulo global, podemos encontrar la línea más larga como:

$$(a'_m, b'_m) = \arg \max_{\substack{0 \leq a' \leq 1 \\ 0 \leq b' \leq 1}} \{h'(a', b')\}$$

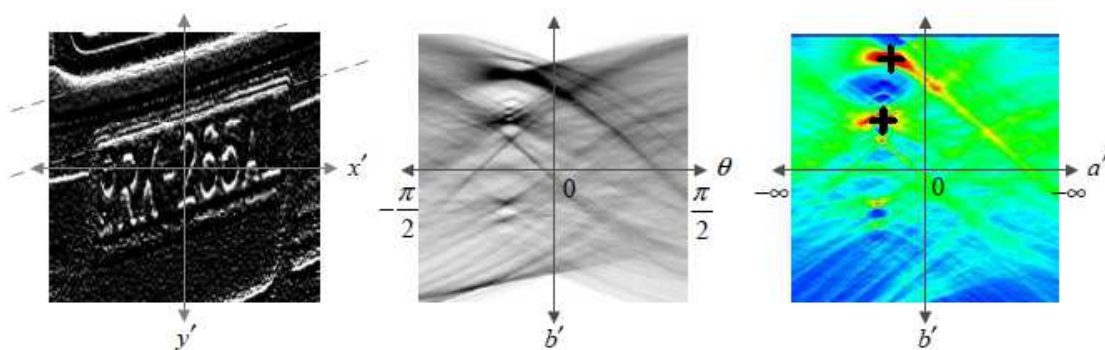


Figura 30. Proceso gráfico de la transformada de Hough.

En la figura anterior, podemos observar el proceso gráfico al aplicar la Transformada de Hough. Primeramente observamos la matrícula en el sistema de coordenadas unificado después aplicar un filtro de detección de borde horizontal. Las dos representaciones siguientes muestran la Transformada de Hough de la matrícula en el sistema de coordenadas “ θB ” y la transformada coloreada en el sistema de coordenadas “ AB ” respectivamente.

Para calcular el ángulo de dicha línea, es necesario para volver a transformarla al sistema de coordenadas original:

$$[a_m, b_m] = \left[w \cdot \frac{a'_m - 1}{2}, h \cdot \frac{b'_m - 1}{2} \right]$$

donde h y w son las dimensiones de la imagen evaluada. Después de esto, el ángulo total de una imagen se puede calcular como:

$$\theta = \arctan(a_m)$$

La solución más sofisticada es determinar el ángulo de la proyección horizontal de la transformada de Hough h' . Esta aproximación es mucho mejor porque cubre todas las líneas paralelas juntas, no solo la más larga:

$$\hat{\theta} = \arctan\left(w \cdot \frac{\hat{a}'_m - 1}{2}\right); \hat{a}'_m = \arg \max_{-1 \leq a' \leq 1} \{p_{a'}(a')\}$$

donde $p_{a'}(a')$ es una proyección horizontal del espacio Hough, tal que:

$$p_{a'}(a') = \int_{-1}^1 f'(a', b') db'$$

4.5.2 Corrección de la desviación

El segundo paso del mecanismo de corrección es una operación geométrica sobre una imagen $f(x, y)$. Como la detección de la desviación se basa en la transformada de Hough, y ésta no logra distinguir entre la desviación y la rotación de una imagen, es importante elegir la operación de enderezado apropiada. En la práctica, las placas están distorsionadas en vez de rotadas en la mayoría de los casos. Para corregir la placa distorsionada por un ángulo θ , usamos la transformación afín para a su vez distorsionarla por el ángulo negativo $-\theta$.

Para esta transformación, se define una matriz de transformación **A**:

$$\mathbf{A} = \begin{bmatrix} 1 & S_y & 0 \\ S_x & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -\tan(\theta) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

donde S_x y S_y son los factores de desviación. El parámetro S_x siempre es cero, porque solo desviamos la placa en la dirección del eje Y.

\mathbf{P} es un vector que representa cierto punto, tal que $\mathbf{P} = [x, y, 1]$ donde x e y son las coordenadas de ese punto. Las nuevas coordenadas $\mathbf{P}_s = [x_s, y_s, 1]$ de ese punto después del procesado, pueden ser calculadas como:

$$f_s(x, y) = f([x, y, 1] \cdot \mathbf{A} \cdot [1, 0, 0]^T, [x, y, 1] \cdot \mathbf{A} \cdot [0, 1, 0]^T)$$

Introduciendo la matriz de transformación \mathbf{A} en la ecuación anterior, obtendríamos la ecuación global:

$$f_s(x, y) = f\left([x, y, 1] \cdot \begin{bmatrix} 1 & -\tan(\theta) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, [x, y, 1] \cdot \begin{bmatrix} 1 & -\tan(\theta) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}\right)$$

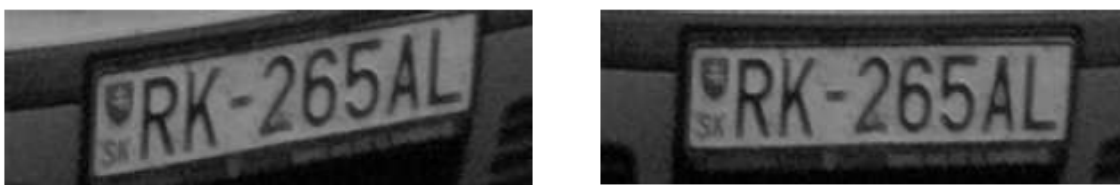


Figura 31. (a) Matrícula original y (b) Matrícula después del proceso de enderezado.

4.5.3 La transformación proyectiva

La transformada de Hough, relativamente, es el método más eficaz y óptimo, pero no el más liviano y menos costoso. Este método empieza a ser costoso computacionalmente en imágenes con dimensiones de 1000x1000, debido a la cantidad de píxeles que tiene que mover y usar para obtener el ángulo correcto, y eso sin contar si tiene que realizar el proceso para un gran array de matrículas consecutivas.

Debido a esto, en determinadas ocasiones es aconsejable utilizar una *transformación proyectiva*, donde las coordenadas de transformación son definidas para llevar a cabo el proceso.

Si una cámara captura una imagen de un objeto 3D, entonces, en general, habrá una perspectiva de asignación de puntos en el objeto a puntos correspondientes en la imagen. Pares de puntos que están a la misma distancia sobre el objeto, estarán más cerca o más separados en la imagen en función de su distancia a la cámara y su orientación con respecto al plano de la imagen. En una transformación perspectiva, consideramos coordenadas en el dominio 3D

(X, Y, Z) que son arbitrariamente distribuidas en el espacio 3D y mapeadas al plano de la imagen de la cámara. Sin embargo, en estas situaciones en las que los puntos de interés en el objeto pueden ser considerados como limitados (ya sea real o aproximadamente) a un plano y estamos interesados en la asignación de los puntos de estos objetos en un plano de imagen arbitrariamente orientado, la transformación requerida se denomina *proyección*.

La forma de la transformación proyectiva está completamente determinada considerando cómo un cuadrilátero arbitrario en el plano del objeto, se mapea en otro cuadrilátero en el plano de la imagen.

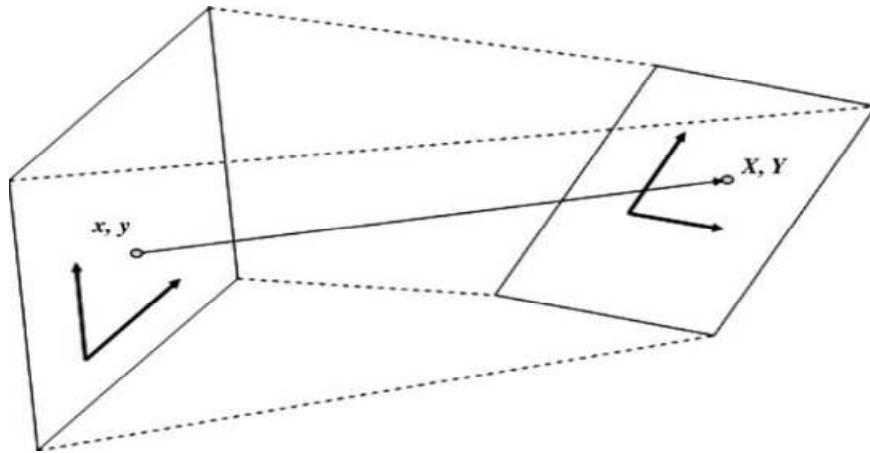


Figura 32. Principio de la transformación proyectiva.

En la figura anterior podemos observar como dadas las coordenadas de las cuatro esquinas de ambos cuadriláteros, se calcula la transformación proyectiva, que asigna un punto arbitrario dentro de un cuadrilátero a su correspondiente punto en el otro cuadrilátero.

Dado que este mapeo está delimitado por cuatro puntos en el espacio 2D, existen ocho coordenadas y, por tanto ocho grados de libertad en una transformación proyectiva. Es posible demostrar que la forma general que la matriz de transformación proyectiva 2D en un sistema de coordenadas homogéneo está dada por:

$$\mathbf{T} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} \end{bmatrix}$$

y las coordenadas transformadas (proyectadas) en el plano de la imagen están relacionadas con los puntos del otro espacio a través de una matriz de multiplicación:

$$\hat{\mathbf{S}} = \mathbf{TS} \rightarrow \begin{bmatrix} x'_1 & \dots & x'_N \\ y'_1 & \dots & y'_N \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} \end{bmatrix} \begin{bmatrix} x_1 & \dots & x_N \\ y_1 & \dots & y_N \\ 1 & 1 & 1 \end{bmatrix}$$

Hay que tener en cuenta que existen ocho grados de libertad en la matriz proyectiva (no nueve), ya que los parámetros están limitados por la relación:

$$\alpha_{31}x + \alpha_{32}y + \alpha_{33} = 1$$



Figura 33. Ejemplos de transformación proyectiva.

La Figura 33. Ejemplos de transformación proyectiva muestra el efecto visual de algunas transformaciones proyectivas. En este caso tenemos un objeto 2D (y por lo tanto plano). Se define un cuadrado en el objeto original con coordenadas $\{0,0; 1,0; 1,1; 0,1\}$ y se exige que se transformen estas coordenadas linealmente a los correspondientes cuadriláteros definidos. La matriz de transformación requerida se calcula invirtiendo la ecuación $\hat{\mathbf{S}}$ sujeta a la restricción expresada por la ecuación de los ocho grados de libertad.

Las transformaciones proyectivas son útiles para registrar o alinear imágenes o, generalmente "escenas" que se pueden aproximar como un plano obtenido a partir de diferentes puntos de vista. La Figura 33. Ejemplos de transformación proyectiva ilustra el uso de la transformación proyectiva como un medio para registrar una imagen con respecto a un punto de vista diferente.

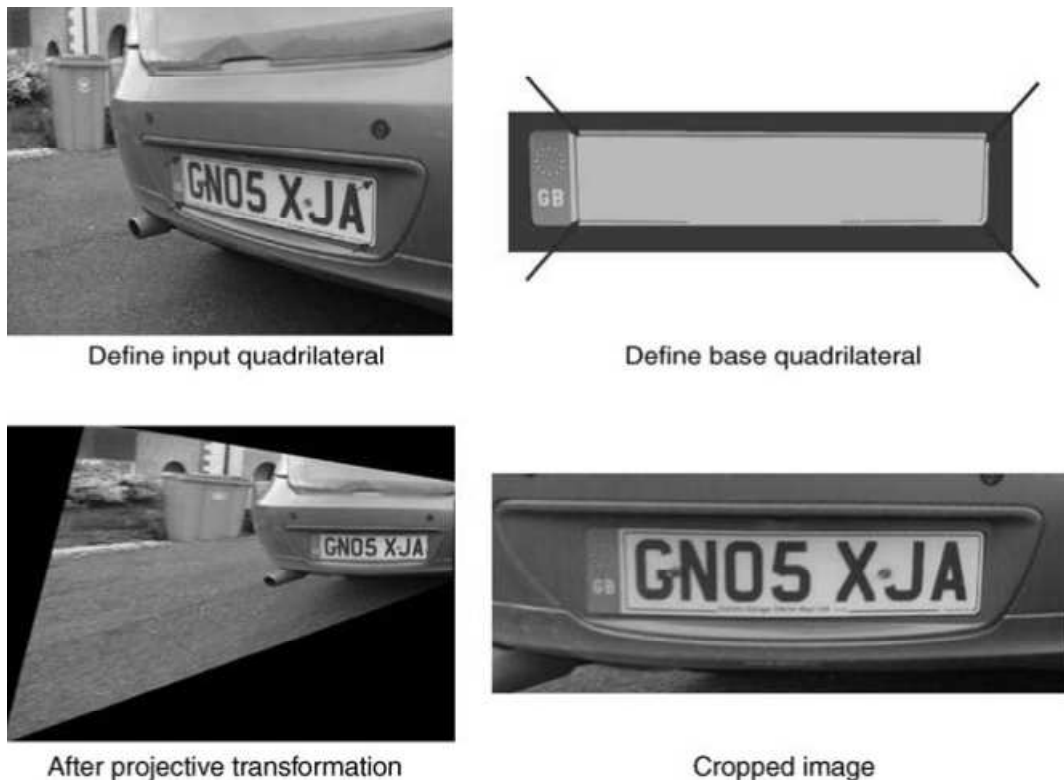


Figura 34. Transformación proyectiva aplicada a una matrícula.

La transformación proyectiva es la *transformación lineal* más común y conserva sólo líneas rectas en la entrada, en la medida que las coordenadas de cualquier punto en la imagen transformada se expresan como una combinación lineal de las coordenadas en la imagen de entrada. Este proceso es especialmente apropiado para situaciones en las que tenemos uno o más ejemplos de una forma o imagen y deseamos alinearlos según un conjunto de reglas o un determinado método.



Figura 35. Ejemplo de transformación proyectiva.

5

Principios de segmentación de caracteres

El paso posterior de la detección del área de la matrícula es la *segmentación de la placa*. La segmentación es una de los procesos más importantes del software de los sistemas ANPR, ya que todos los procesos posteriores se apoyan en éste. Si la segmentación falla, un carácter puede ser incorrectamente dividido en dos piezas, o dos caracteres pueden estar incorrectamente fusionados.

Podemos usar una proyección horizontal de una matrícula para la segmentación, o uno de los métodos más sofisticados, usando redes neuronales.

La segunda fase de la segmentación es un realce de los segmentos. El segmento de una placa contiene, además de los caracteres, elementos indeseables, tales como puntos y extensiones, así como espacio redundante a los lados del carácter; por lo que es necesario eliminarlos y extraer solo el carácter.

5.1 Segmentación de la placa usando una proyección horizontal

Dado que la placa segmentada está correctamente enderezada, podemos segmentarla detectando espacios en su proyección horizontal. En ocasiones se aplica un filtro con umbral adaptativo para realzar un área de la placa antes de la segmentación. El umbral adaptativo es usado para separar el primer plano oscuro del fondo luminoso con iluminación no uniforme.

Después del umbralizado, se calcula la proyección horizontal $p_x(x)$ de la placa $f(x,y)$. Usamos esta proyección para determinar límites horizontales entre caracteres segmentados. Estos límites se corresponden con picos en la gráfica de la proyección horizontal.

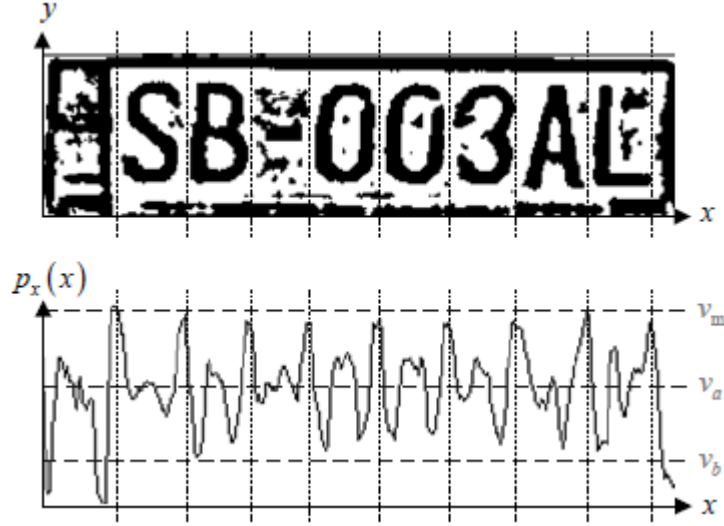


Figura 36. (a) Matrícula después de la aplicación de un umbral adaptativo. (b) Proyección horizontal de la placa con los picos detectados.

El objetivo del algoritmo de segmentación es encontrar picos, que se corresponden a los espacios entre caracteres. Primeramente se necesita definir varios valores importantes en la gráfica de la proyección horizontal $p_x(x)$:

- v_m : Valor máximo contenido en la proyección horizontal $p_x(x)$, tal que $v_m = \max_{0 \leq x < w} \{p_x(x)\}$, donde w es el ancho de la placa en píxeles.
- v_a : Valor medio de la proyección horizontal $p_x(x)$, tal que $v_a = \frac{1}{w} \sum_{x=0}^{w-1} p_x(x)$.
- v_b : Este valor es usado como base para la evaluación de la altura del pico. El valor base es siempre calculado como $v_b = 2 \cdot v_a - v_m$. La variable v_a debe estar en el eje vertical, entre los valores v_b y v_m .

El algoritmo de segmentación encuentra iterativamente el pico máximo en la gráfica de la proyección vertical. El pico es tratado como un espacio entre caracteres si éste satisface algunas condiciones adicionales, tal como la altura del pico. Por tanto el algoritmo detecta el pico e iterativamente repite este proceso hasta que no se encuentran más espacios. El proceso está formado por los siguientes pasos:

1. Determinar el índice del valor máximo de la proyección horizontal:

$$x_m = \arg \max_{0 \leq x < w} \{p_x(x)\}$$
2. Detectar la base izquierda y derecha del pico determinado:

$$x_l = \max_{0 \leq x \leq x_m} \{x | p_x(x) \leq c_x \cdot p_x(x_m)\}$$

$$x_r = \max_{x_m \leq x < w} \{x | p_x(x) \leq c_x \cdot p_x(x_m)\}$$
3. Realizar la proyección horizontal $p_x(x)$ en el intervalo $\langle x_l, x_r \rangle$

4. Si $p_x(x_m) < c_w \cdot v_m$, ir al paso 7.
5. Dividir la placa horizontalmente en el punto x_m .
6. Ir al paso 1.
7. Fin.

Dos diferentes constantes han sido usadas en el algoritmo explicado. La constante c_x es usada para determinar las bases del pico x_m . El valor óptimo de c_x es 0,7.

La constante c_w determina la altura mínima del pico relacionada con el valor máximo de la proyección (v_m). Si la altura del pico está por debajo de este mínimo, el pico no será considerado como un espacio entre caracteres. Es importante escoger un valor para la constante c_w cuidadosamente. Un inadecuado pequeño valor puede causar demasiados picos que serán tratados como espacios, y los caracteres serán impropriamente divididos. Un gran valor de c_w causa que no todos los picos regulares sean tratados como espacios, y los caracteres serán incorrectamente juntados o fusionados. El valor óptimo de c_w es 0,86. Para asegurar un comportamiento apropiado del algoritmo, las constantes c_x y c_w deberían cumplir la siguiente condición:

$$\forall (x_l, x_m, x_r) \in P: c_w \cdot v_m > p_x(x_l) \cap p_x(x_r)$$

Donde **P** es el conjunto de todos los picos detectados x_m con sus correspondientes bases x_l y x_r .

5.2 Extracción de los caracteres de los segmentos horizontales

El segmento de la placa contiene además de los caracteres, espacio redundante y otros elementos indeseables. Entendemos que bajo el término “segmento”, la parte de una matrícula determinada por un algoritmo de segmentación horizontal. Ya que el segmento ha sido procesado por un filtro de umbral adaptativo, éste solo contiene píxeles blancos y negros. Los píxeles vecinos están agrupados de forma conjunta en piezas más grandes, y una de ellas es un carácter. Nuestro objetivo es dividir el segmento en varias piezas y mantener solamente una de las piezas que representa el carácter que buscamos. Este concepto está ilustrado en la figura 37.

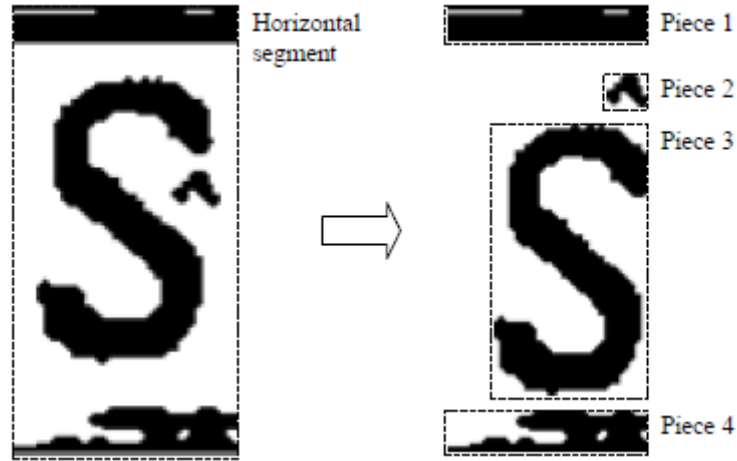


Figura 37. El segmento horizontal de la matrícula contiene varios grupos (piezas) de píxeles vecinos.

5.2.1 Extracción de las piezas

Tenemos un segmento definido por una función discreta $f(x, y)$ en un relativo sistema de coordenadas, tal que $[0, 0]$ es la esquina superior izquierda del segmento, y $[w - 1, h - 1]$ es la esquina inferior derecha, donde w y h son las dimensiones del segmento. El valor de $f(x, y)$ es “1” para los píxeles negros, y “0” para los espacios en blanco.

La pieza \mathbf{P} es un conjunto de todos los píxeles vecinos $[x, y]$, que representan un elemento continuo. El píxel $[x, y]$ pertenece a la pieza \mathbf{P} si hay al menos un píxel $[x', y']$ de \mathbf{P} , tal que $[x, y]$ y $[x', y']$ sean vecinas:

$$[x, y] \in \mathbf{P} \leftrightarrow \exists [x', y'] \in \mathbf{P}: [x, y] \check{N}_4 [x', y']$$

La notación $a \check{N}_4 b$ significa una relación binaria “ a es vecina de b por una vecindad de 4 píxeles”:

$$[x, y] \check{N}_4 [x', y'] \leftrightarrow |x - x'| = 1 \oplus |y - y'| = 1$$

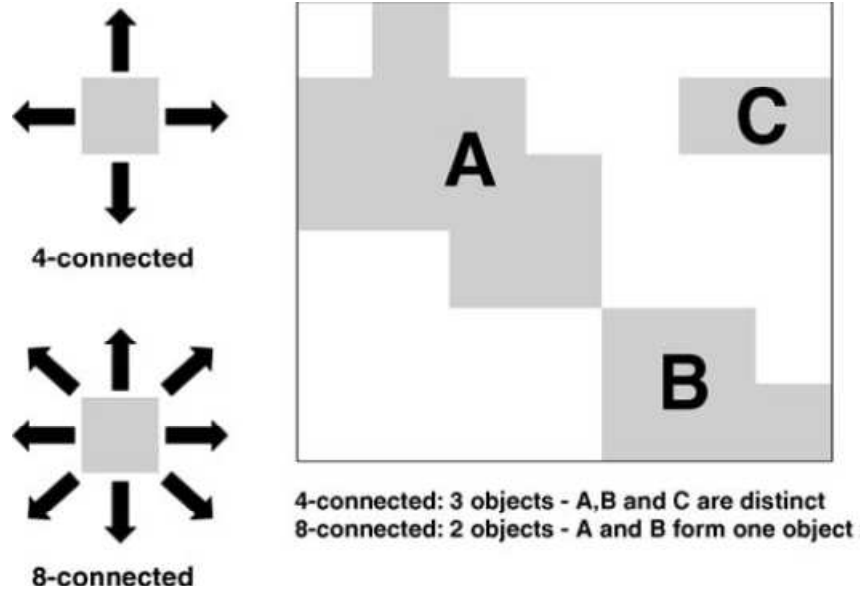


Figura 38. Ejemplo de conexión de 4 píxeles u 8 píxeles.

5.2.2 Análisis heurístico de las piezas

La pieza es un conjunto de píxeles en el sistema local de coordenadas del segmento. El segmento usualmente contiene varias piezas. Una de ellas representa el carácter y las otras los elementos redundantes, que deberían ser eliminados. El objetivo del análisis heurístico es encontrar una pieza, la cual representa el carácter.

Sea una pieza P dentro de un rectángulo imaginario (x_0, y_0, x_1, y_1) , donde $[x_0, y_0]$ es la esquina superior izquierda, y $[x_1, y_1]$ la esquina inferior derecha de la pieza:

$$x_0 = \min \{x | [x, y] \in P\}, \quad y_0 = \min \{y | [x, y] \in P\}$$

$$x_1 = \max \{x | [x, y] \in P\}, \quad y_1 = \max \{y | [x, y] \in P\}$$

Las dimensiones y el área imaginaria del rectángulo imaginario son definidas como $w = |x_0 - x_1|$, $h = |y_0 - y_1|$ y $S = w \cdot h$. La cardinalidad del conjunto P representa el número de píxeles negros n_b . El número de píxeles blancos n_w puede ser calculado como $n_w = S - n_b = w \cdot h - |P|$. La magnitud global M de una pieza es la proporción entre el número de píxeles negros n_b y el área S de un rectángulo imaginario, tal que $M = n_b/S$.

En la práctica, usamos el número de píxeles blancos n_w como un heurístico. Las piezas con un alto valor de n_w serán las preferentes.

La pieza elegida por la heurística es entonces convertida a una imagen monocroma. Cada imagen corresponde a un segmento horizontal. Estas imágenes están consideradas como la salida de la fase de segmentación del proceso ANPR.

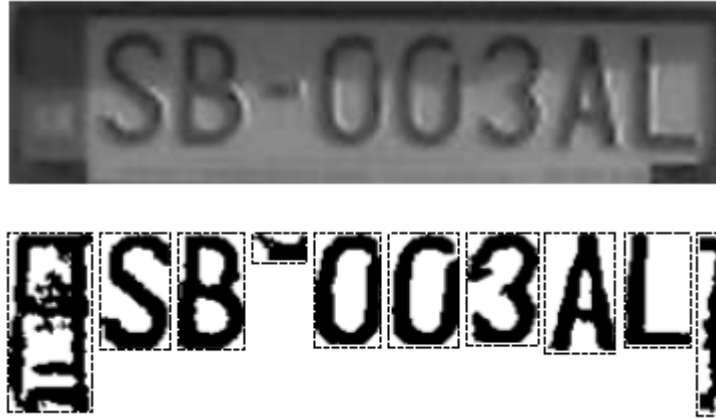


Figura 39. Imagen de entrada y salida después de la fase de segmentación.

6 Extracción de características y normalización de caracteres

Para reconocer un carácter de una representación en un plano, se necesita extraer los descriptores de características de tales planos. Como un método de extracción afecta a la calidad de todo el proceso OCR, es muy importante extraer las características, que serán invariantes hacia las diversas condiciones de luz, el tipo de fuente empleada y las deformaciones en los caracteres causadas por la desviación de la imagen.

El primer paso es una normalización del brillo y el contraste de los segmentos de imagen procesados. Los caracteres que figuran en los segmentos de la imagen deben ser redimensionados para unificar las dimensiones. Después de esto, el algoritmo de extracción de características extrae los descriptores apropiados de los caracteres normalizados. Este capítulo trata sobre varios métodos utilizados en el proceso de normalización.

6.1 Normalización del brillo y el contraste

Las características de brillo y contraste de los caracteres segmentados son variables, debido a las diferentes condiciones de luz durante la captura de la imagen. Debido a esto, es necesario normalizarlos. Existen diferentes vías, pero vamos a describir las tres más usadas: *normalización del histograma*, *umbralización global* y *umbralización adaptativa*.

A través de la normalización del histograma, las intensidades de los segmentos del carácter son redistribuidos en el mismo, para obtener estadísticas normalizadas.

Las técnicas de umbralización global y adaptativa se utilizan para obtener las representaciones monocromas de los segmentos de carácter procesados. La representación monocroma de la imagen es más apropiada para su análisis, ya que define claramente los límites contenidos en los caracteres.

6.1.1 Normalización del histograma

La normalización del histograma es un método utilizado para redistribuir los intensidades de brillo dentro del histograma del segmento de caracteres. Las áreas de menos contraste ganarán un mayor contraste sin afectar las características globales de la imagen.

Considerando una imagen en escala de grises definida por una función discreta $f(x,y)$. Sea I el número total de niveles de gris en la imagen (por

ejemplo $I=256$). Usamos un histograma para determinar el número de ocurrencias de cada nivel de gris i , $i \in 0 \dots I - 1$:

$$H(i) = |\{[x, y] | 0 \leq x < w \cap 0 \leq y < h \cap f(x, y) = i\}|$$

Los valores mínimo, máximo y medio contenido en el histograma se definen como:

$$H_{min} = \min_{\substack{0 \leq x < w \\ 0 \leq y < h}} \{f(x, y)\} ; H_{max} = \max_{\substack{0 \leq x < w \\ 0 \leq y < h}} \{f(x, y)\} ; H_{medio} = \frac{1}{w \cdot h} \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} f(x, y)$$

donde los valores H_{min} , H_{max} y H_{medio} se sitúan de la siguiente manera:

$$0 \leq H_{min} \leq H_{medio} \leq H_{max} \leq I - 1$$

El objetivo de la normalización del histograma consiste en obtener una imagen con características, tal que $H_{min} = 0, H_{max} = I - 1, H_{medio} = \frac{I}{2}$. Para cumplir este objetivo, se construye una función de transformación $g(i)$ como un polinomio de Lagrange con puntos de interpolación $[x_1, y_1] = [H_{min}, 0], [x_2, y_2] = [H_{medio}, \frac{I}{2}] \cap [x_3, y_3] = [H_{max}, I - 1]$:

$$g(i) = \sum_{j=1}^3 \left(y_j \prod_{\substack{k=1 \\ k \neq j}}^3 \frac{i - x_k}{x_j - x_k} \right)$$

La función de integración puede ser escrita explícitamente como:

$$g(i) = y_1 \cdot \frac{i - x_2}{x_1 - x_2} \cdot \frac{i - x_3}{x_1 - x_3} + y_2 \cdot \frac{i - x_1}{x_2 - x_1} \cdot \frac{i - x_3}{x_2 - x_3} + y_3 \cdot \frac{i - x_1}{x_3 - x_1} \cdot \frac{i - x_2}{x_3 - x_2}$$

Después de la sustitución de los punto concretos y de un número concreto de niveles grises $I = 256$:

$$g(i) = +128 \cdot \frac{i - H_{min}}{H_{medio} - H_{min}} \cdot \frac{i - H_{max}}{H_{medio} - H_{max}} + 255 \cdot \frac{i - H_{min}}{H_{max} - H_{min}} \cdot \frac{i - H_{medio}}{H_{max} - H_{medio}}$$

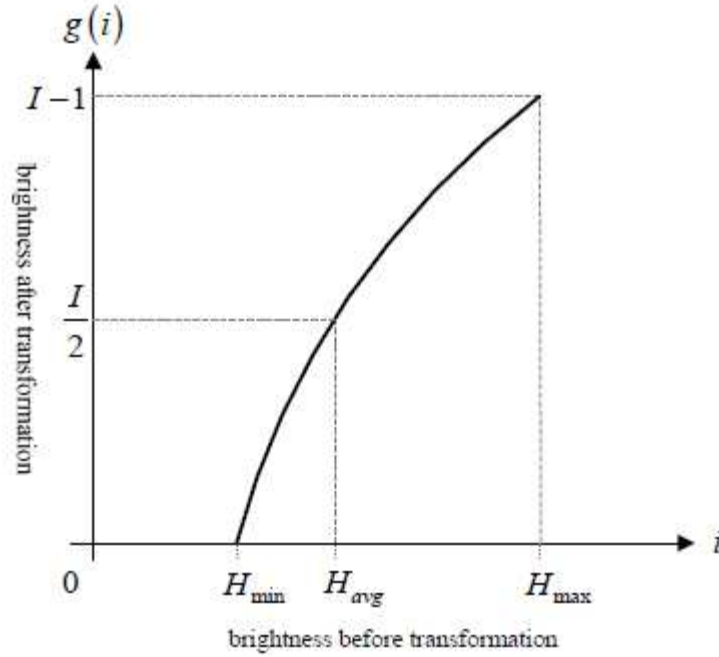


Figura 40. Utilizamos una interpolación polinomial de Lagrange como una función de transformación.

La interpolación polinomial de Lagrange como una transformación es una solución costosa. En la práctica, es más útil construir la transformación usando una simple función lineal que se extiende el intervalo $\langle H_{min}, H_{max} \rangle$ dentro de un intervalo unificado $\langle 0, I - 1 \rangle$:

$$g(i) = \frac{i - H_{min}}{H_{max} - H_{min}}(I - 1)$$

La normalización de la imagen está procedida por la función de transformación de la siguiente manera:

$$f_n(x, y) = g(f(x, y))$$

6.1.2 Umbralización global

La umbralización global es una operación, en la que una escala de grises continua de una imagen, es reducida dentro de colores monocromos de acuerdo al valor de un umbral global. Sea 0 y 1 la escala de grises de dicha imagen. Si el valor de un cierto pixel está por debajo de un umbral t , el nuevo valor del pixel será cero. En caso contrario, el nuevo valor será uno para los pixeles con valores por encima del umbral t .

Sea v un valor original del pixel, tal que $v \in \langle 0, 1 \rangle$. El nuevo valor v' se calcula como:

$$v' = \begin{cases} 0 & \text{si } v \in \langle 0, t \rangle \\ 1 & \text{si } v \in \langle t, 1 \rangle \end{cases}$$

El valor de umbral t se puede obtener usando una aproximación heurística, basada en una inspección visual del histograma. Usamos el siguiente algoritmo para determinar el valor de t automáticamente:

1. Seleccionar un *umbral inicial* estimado t , por ejemplo $t=0.5$.
2. El umbral t divide los pixeles en dos diferentes conjuntos:
 $S_a = \{[x, y] | f(x, y) < t\}$, y $S_b = \{[x, y] | f(x, y) \geq t\}$.
3. Calcular el *valor medio* de los niveles de gris S_a y S_b , para los conjuntos S_a y S_b como:
 $\mu_a = \frac{1}{|S_a|} \sum_{[x,y] \in S_a} f(x, y)$; $\mu_b = \frac{1}{|S_b|} \sum_{[x,y] \in S_b} f(x, y)$
4. Calcular un nuevo valor umbral $t = \frac{1}{2}(\mu_a + \mu_b)$.
5. Repetir los pasos 2, 3, 4 hasta que la diferencia Δt en sucesivas iteraciones sea más pequeña que el umbral predefinida t_p .

Dado que el umbral t es global para toda la imagen, este método puede fallar en ocasiones. La Figura 41. Resultado erróneo de una umbralización global muestra una matrícula parcialmente sombreada. Si calculamos el umbral t usando el algoritmo previamente descrito, todos los pixeles de la parte sombreada estarán por debajo de este umbral y todos los otros pixeles estarán por encima. Esto ocasiona un resultado no deseado que se muestra a continuación.

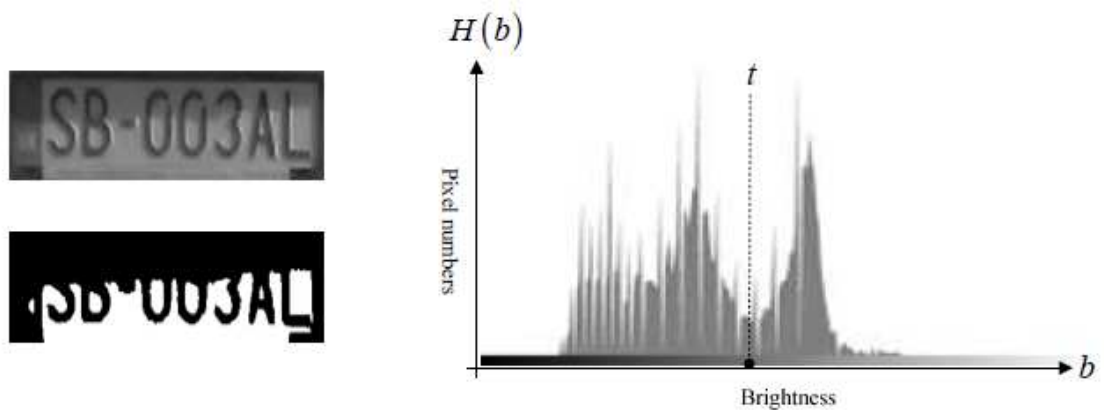


Figura 41. Resultado erróneo de una umbralización global.

6.1.3 Umbralización adaptativa

La matrícula en ocasiones puede estar parcialmente sombreada o no uniformemente iluminada. Esta es la razón más común, por la que la umbralización global falla. La umbralización adaptativa resuelve varias desventajas de la umbralización global, ya que calcula el valor de umbral para cada pixel independientemente tomando como referencia sus pixeles de alrededor.

6.1.3.1 Aproximación de Chow y Kaneko

Existen dos aproximaciones para calcular este tipo de umbral. La primera es la *aproximación de Chow y Kaneko*, y la otra es la *umbralización local*. Ambos métodos asumen que regiones rectangulares más pequeñas, son más propensas a tener una iluminación aproximadamente uniforme, más adecuada para la umbralización. La imagen se divide en áreas rectangulares uniformes con un tamaño de $m \times n$ píxeles. El histograma local se calcula para cada área y a su vez se determina un umbral local. El umbral de un punto concreto es entonces calculado por la interpolación de los resultados de las subimágenes.

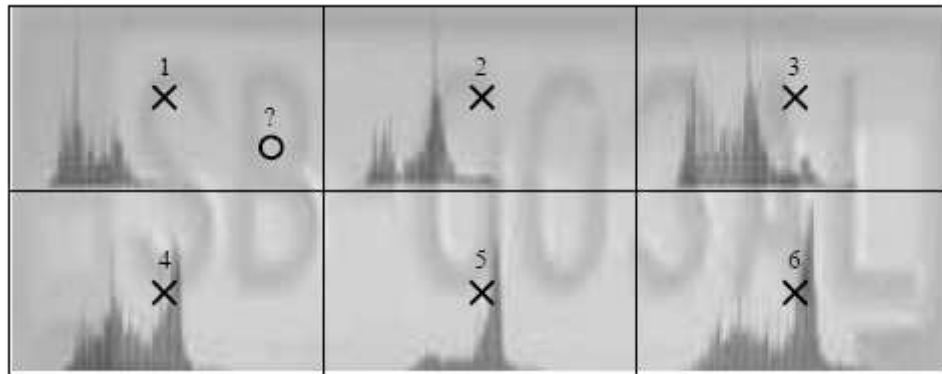


Figura 42. Matrícula procesada por la aproximación de Chow y Kaneko.

Como podemos observar en la figura anterior, la matrícula se divide en varias áreas, cada una de ellas con su propio histograma y valor umbral. El valor umbral de un pixel concreto (denotado por un círculo en la figura), se calcula por una interpolación de los resultados de las subimágenes colindantes (representada por los pixeles del 1 al 6).

6.1.3.2 Umbralización local

El segundo método para obtener el umbral local del pixel, consiste en un examen estadístico de los píxeles vecinos. Sea $[x,y]$ un pixel, para el cual calculamos un umbral local t . Para simplificar podemos considerar una

vecindad cuadrada con un ancho de $2 \cdot r + 1$, donde $[x - r, y - r]$, $[x - r, y + r]$, $[x + r, y - r]$ y $[x + r, y + r]$ son las esquinas de dicho cuadrado. Existen varias aproximaciones de calcular el valor del umbral:

- *Media de la vecindad:* $t(x, y) = \text{media}_{\substack{x-r \leq i \leq x+r \\ y-r \leq j \leq y+r}} \{f(i, j)\}$
- *Mediana de la vecindad:* $t(x, y) = \text{mediana}_{\substack{x-r \leq i \leq x+r \\ y-r \leq j \leq y+r}} \{f(i, j)\}$
- *Media del valor máximo y mínimo de la vecindad:*

$$t(x, y) = \frac{1}{2} \left(\min_{\substack{x-r \leq i \leq x+r \\ y-r \leq j \leq y+r}} \{f(i, j)\} + \max_{\substack{x-r \leq i \leq x+r \\ y-r \leq j \leq y+r}} \{f(i, j)\} \right)$$

El nuevo valor $f'(x, y)$ del pixel $[x, y]$ es entonces calculado como:

$$f'(x, y) = \begin{cases} 0 & \text{si } f(x, y) \in \langle 0, t(x, y) \rangle \\ 1 & \text{si } f(x, y) \in \langle t(x, y), 1 \rangle \end{cases}$$

6.2 Normalización de las dimensiones y remuestreo

Antes de la extracción de los descriptores de características de la representación de la imagen (o mapa de bits) de un carácter, es necesario normalizarla dentro de unas dimensiones predefinidas. Se entiende bajo el término de “remuestreo” al proceso de modificar o redimensionar las dimensiones de un carácter. Como las dimensiones de los caracteres sin normalizar son usualmente mayores que los normalizados, en la mayoría de los casos se realizará un proceso de diezmado. Cuando diezmamos muestras, reducimos la información contenida en la imagen procesada.

Existen varios métodos de remuestreo, tales como el *redimensionamiento de pixel*, la *interpolación bilinear* o el *remuestreo promedio ponderado*. No podemos determinar cuál es el mejor método en general, ya que el éxito de un método en particular depende de muchos factores. Por ejemplo, el uso del diezmado promedio ponderado combinado con un filtro de detección de bordes no es una buena solución, ya que este tipo de diezmado no conserva la nitidez o un alto contraste en los bordes (más tarde desarrollado). Debido a esto, la problemática del remuestreo de caracteres está estrechamente asociada con la problemática de la extracción de características.

Para el desarrollo de este apartado, vamos a suponer que $m \times n$ son las dimensiones de la imagen original, y $m' \times n'$ son las dimensiones de la imagen después del remuestreo. La relación horizontal y vertical estará definida como $r_x = m'/m$ y $r_y = n'/n$, respectivamente.

6.2.1 Diezmado de vecindad más cercana

El principio de *diezmado de vecindad más cercana* consiste en coger el pixel más cercano en la imagen original que corresponda a un pixel procesado en la imagen remuestreada. Sea $f(x, y)$ una función discreta que defina la imagen original, tal que $0 \leq x < n$. Por lo tanto, la función $f'(x', y')$ de la imagen después del remuestro se define como:

$$f'(x', y') = f\left(\left\lfloor \frac{x'}{r_x} \right\rfloor, \left\lfloor \frac{y'}{r_y} \right\rfloor\right)$$

donde $0 \leq x' < m'$ y $0 \leq y' < n'$.

Si la relación de aspecto es menor que uno, cada pixel en la nueva imagen muestreada (o imagen destino) corresponde a un grupo de píxeles en la imagen original, pero sólo un valor de entre el grupo de píxeles origen influye en el valor del pixel de la imagen remuestreada (en este caso diezmada). Este hecho provoca una significativa reducción de la información contenida en la imagen original (ver Figura 44. Resultado a partir del método de *diezmado de pixel vecino más cercano* y por *diezmado promedio ponderado* respectivamente.

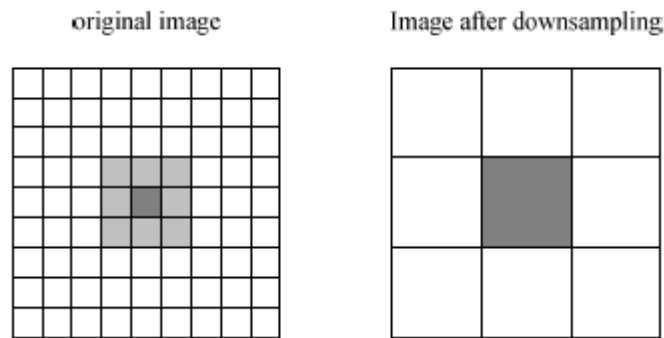


Figura 43. Un pixel en la imagen remuestreada corresponde a un grupo de píxeles en la imagen original.

Aunque el diezmado de los vecinos más cercanos reduce significativamente la información contenida en la imagen original, ignorando así una gran cantidad de píxeles, éste conserva el contraste en los bordes y la fuerte bipolaridad de los píxeles blancos y negros. Debido a esto, el diezmado del vecino más cercano es adecuado en combinación con la extracción de

características de “detección de bordes”, que será descrito y explicado más adelante.

6.2.2 Diezmado promedio ponderado

A diferencia del método del vecino más cercano, el *diezmado promedio ponderado* considera todos los píxeles de un determinado grupo en la imagen original.

Sea r_x y r_y la relación de aspecto de la imagen remuestreada. El valor del pixel $[x', y']$ en la imagen destino se tratará de la media de los píxeles origen dentro del rango $[x_{min}, y_{min}]$ a $[x_{max}, y_{max}]$:

$$f'(x', y') = \frac{1}{(x_{max} - x_{min}) \cdot (y_{max} - y_{min})} \sum_{i=x_{min}}^{x_{max}} \sum_{j=y_{min}}^{y_{max}} f(i, j)$$

donde:

$$x_{min} = \left\lfloor \frac{x'}{r_x} \right\rfloor; y_{min} = \left\lfloor \frac{y'}{r_y} \right\rfloor; x_{max} = \left\lfloor \frac{x' + 1}{r_x} \right\rfloor; y_{max} = \left\lfloor \frac{y' + 1}{r_y} \right\rfloor;$$

El método de diezmado por promedio ponderado no conserva la nitidez en los bordes de la imagen (al contrario que el método anteriormente descrito). Se puede observar una comparación visual de estos dos métodos en la siguiente la Figura 44. Resultado a partir del método de *diezmado de pixel vecino más cercano* y por *diezmado promedio ponderado* respectivamente.

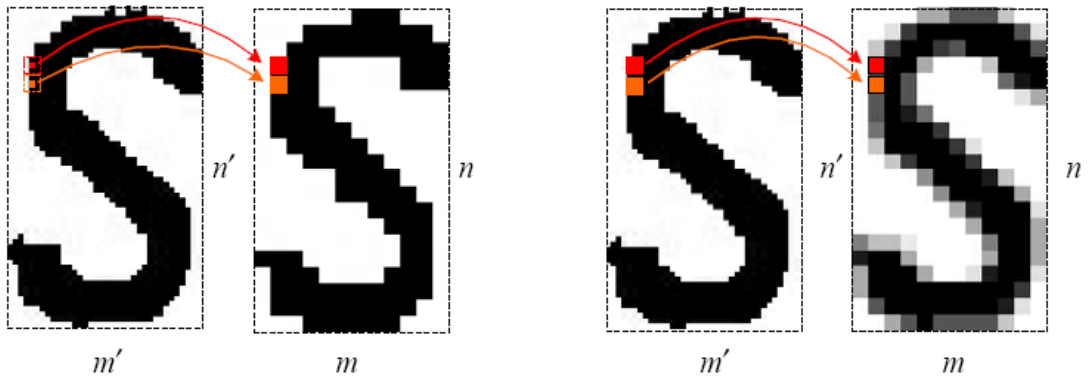


Figura 44. Resultado a partir del método de *diezmado de pixel vecino más cercano* y por *diezmado promedio ponderado* respectivamente.

6.3 Extracción de características

La información contenida en el mapa de bits de una imagen no es adecuada para el procesamiento digital. Debido a esto, es necesario describir un carácter de otra manera. La descripción del carácter debería ser invariable hacia el tipo de fuente usado, o deformaciones causadas por el proceso de centrado. Además, en todos los casos del mismo carácter debería tener una descripción similar. Una descripción del carácter consiste en un vector de valores numerales, llamados “descriptores” o “patrones”:

$$X = (x_0, \dots, x_{n-1})$$

Generalmente, la descripción de una región de la imagen está basada en su representación interna y externa. La representación interna está basada en sus propiedades regionales, como el color o textura. La descripción de caracteres normalizados está basada en sus características externas, ya que solo afectan a propiedades como la forma del carácter. Por tanto, el vector de descriptores incluye características como el número de líneas, franjas, cantidad de líneas horizontales, verticales y diagonales, bordes, etc. La extracción de características es un proceso de transformación de los datos de una representación en mapa de bits en una serie de descriptores, que son más adecuados para el procesamiento digital.

Si asociamos casos similares del mismo carácter dentro de distintas clases, entonces los descriptores de los caracteres de la misma clase deberían ser geoméricamente similares entre ellos en el vector espacio. Esto es un supuesto básico para el éxito en el proceso de reconocimiento de patrones.

Esta sección se ocupa de los distintos métodos de extracción de características, y explica cual es el método más adecuado para un tipo específico de mapa de bit de un determinado carácter.

6.3.1 La matriz de píxeles

La manera más simple de extraer los descriptores de un mapa de bits es asignar un brillo a cada pixel con un correspondiente valor en el vector de descriptores. Por lo tanto, la longitud de cada sector es igual a un cuadrado ($w \cdot h$) del mapa de bits transformado:

$$x_i = f\left(\left\lfloor \frac{i}{w} \right\rfloor, \text{mod}_w(i)\right)$$

donde $i \in 0, \dots, w \cdot h - 1$.

Mapas de bits más grandes producen vectores extremadamente largos, que no son aceptables para el reconocimiento. Debido a esto, el tamaño de tal mapa de bit procesado es muy limitado. Además, este método no considera la cercanía geométrica de los píxeles, así como sus relaciones. Dos casos parecidos del mismo carácter en muchas ocasiones producen vectores de descripción muy diferentes. A pesar de esto, este método es el adecuado si los mapas de bits de los caracteres son demasiado borrosos o demasiado pequeños para la detección de bordes.

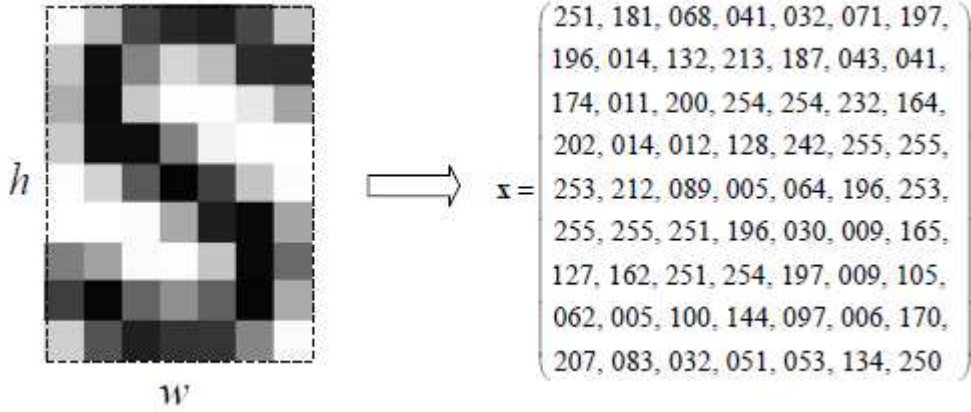


Figura 45. Ejemplo de matriz de píxeles para la extracción de características.

6.3.2 Detección de bordes del carácter

Al contrario que el método anterior, la detección de bordes no considera en absoluto el posicionamiento de cada pixel, pero solamente en un número de tipos de borde individuales en una región específica del mapa de bits del carácter. Debido a esto, el vector resultante es invariable hacia el desplazamiento intra-regional de los bordes y hacia pequeñas deformaciones de los caracteres.

6.3.2.1 Regiones del mapa de bits

Sea un mapa de bits descrito por una función discreta $f(x, y)$, donde w y h son sus dimensiones, tales que $0 \leq x < w$ y $0 \leq y < h$. Lo dividimos en seis regiones iguales organizadas en tres filas y dos columnas de la siguiente manera:

Sea $[x_{min}^i, y_{min}^i]$ y $[x_{max}^i, y_{max}^i]$ un punto superior izquierdo y otro inferior derecho de un rectángulo, que determinan la región r_i , tal que:

- Región r_0 : $x_{min}^{(0)} = 0$, $y_{min}^{(0)} = 0$, $x_{max}^{(0)} = \left\lceil \frac{w}{2} \right\rceil - 1$, $y_{max}^{(0)} = \left\lceil \frac{h}{3} \right\rceil - 1$

- Región r_1 : $x_{min}^{(1)} = \left\lceil \frac{w}{2} \right\rceil$, $y_{min}^{(1)} = 0$, $x_{max}^{(1)} = w - 1$, $y_{max}^{(1)} = \left\lceil \frac{h}{3} \right\rceil - 1$
- Región r_2 : $x_{min}^{(2)} = 0$, $y_{min}^{(2)} = \left\lceil \frac{h}{3} \right\rceil$, $x_{max}^{(2)} = \left\lceil \frac{w}{2} \right\rceil - 1$, $y_{max}^{(2)} = \left\lceil \frac{2 \cdot h}{3} \right\rceil - 1$
- Región r_3 : $x_{min}^{(3)} = \left\lceil \frac{w}{2} \right\rceil$, $y_{min}^{(3)} = \left\lceil \frac{h}{3} \right\rceil$, $x_{max}^{(3)} = w - 1$, $y_{max}^{(3)} = \left\lceil \frac{2 \cdot h}{3} \right\rceil - 1$
- Región r_4 : $x_{min}^{(4)} = 0$, $y_{min}^{(4)} = \left\lceil \frac{2 \cdot h}{3} \right\rceil$, $x_{max}^{(4)} = \left\lceil \frac{w}{2} \right\rceil$, $y_{max}^{(4)} = h - 1$
- Región r_5 : $x_{min}^{(5)} = \left\lceil \frac{w}{2} \right\rceil$, $y_{min}^{(5)} = \left\lceil \frac{2 \cdot h}{3} \right\rceil$, $x_{max}^{(5)} = w - 1$, $y_{max}^{(5)} = h - 1$

Hay varios caminos para distribuir las regiones en el mapa de bits. Las regiones pueden ser disyuntivas, así como se pueden solapar unas con otras. La siguiente figura muestra los diversos diseños de las regiones.

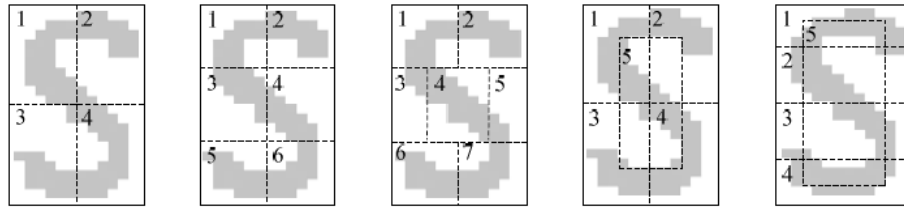


Figura 46. Diseños de las regiones en el mapa de bits del carácter.

6.3.2.2 Tipo de bordes de las regiones

Sea un borde del carácter definido como una transición 2x2 de blanco a negro en un mapa de bits. De acuerdo con esta definición, la imagen puede contener catorce tipos diferentes de bordes, ilustrados en la figura 28.

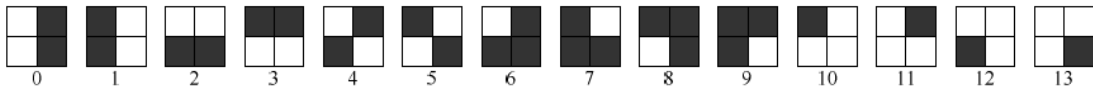


Figura 47. El mapa de bits procesado puede contener diferentes tipos de bordes de dimensiones 2x2.

Las estadísticas de cada tipo de borde hace inútil el vector de descriptores. Debido a esto, tipos similares de bordes son considerados iguales. Las siguientes listas, muestran como pueden ser agrupados:

- 0+1 (bordes verticales)
- 2+3 (bordes horizontales)
- 4+6+9 ("/" bordes diagonales)

- 5+7+8 (“\” bordes diagonales)
- 10 (esquina superior izquierda)
- 11 (esquina superior derecha)
- 12 (esquina inferior izquierda)
- 13 (esquina inferior derecha)

Por simplicidad, suponemos que los tipos de borde no están agrupados. Sea η un número diferente de tipos de borde, donde h_i es una matriz de 2x2 que se corresponde con un tipo específico de borde:

$$h_0 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, h_1 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, h_2 = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, h_3 = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}, h_4 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, h_5 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, h_6 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

$$h_7 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, h_8 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, h_9 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, h_{10} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, h_{11} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, h_{12} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, h_{13} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Sea ρ un número de regiones rectangulares en el mapa de bits del carácter, donde $x_{\min}^{(i)}$, $y_{\min}^{(i)}$, $x_{\max}^{(i)}$, $y_{\max}^{(i)}$ son límites de la región r_i ($i \in 0 \dots \rho - 1$). Si estadísticamente consideramos η diferentes tipos de borde para cada ρ regiones, la longitud del vector resultante x se calcula como $\eta \cdot \rho$:

$$x = (x_0, x_1, \dots, x_{\eta \cdot \rho - 1})$$

6.3.2.3 Algoritmo de extracción de características

En primer lugar, tenemos que integrar el mapa de bits $f(x, y)$ en uno mayor con relleno de color blanco, para asegurar el comportamiento adecuado del algoritmo. Sea el relleno de un pixel de ancho, por tanto, las dimensiones del mapa de bits incrustado o embebido, serán $w + 2$ e $y + 2$. El mapa de bits embebido $f'(x, y)$ se define como:

$$f'(x, y) = \begin{cases} 1 & \text{si } (x = 0 \cup y = 0 \cup x = w + 1 \cup y = h + 1) \\ f(x - 1, y - 1) & \text{si } \text{No}(x = 0 \cup y = 0 \cup x = w + 1 \cup y = h + 1) \end{cases}$$

donde w y h son las dimensiones del mapa de bits antes del proceso de embebido. El color del relleno es blanco (valor 1). Las coordenadas de los píxeles se desplazan un pixel hacia la posición original.

La estructura del vector de descriptores de salida está ilustrado por el patrón mostrado más abajo. La notación $h_j @ r_i$ significa “el número de casos de un borde representado por la matriz h_j en la región r_i ”.

$$x = (h_0 @ r_0, h_1 @ r_0, \dots, h_{\eta-1} @ r_0; h_0 @ r_1, h_1 @ r_1, \dots, h_{\eta-1} @ r_1; h_0 @ r_{\rho-1}, h_1 @ r_{\rho-1}, \dots, h_{\eta-1} @ r_{\rho-1})$$

Calculamos la posición k de $h_j@r_i$ en el vector x como $k = i \cdot \eta + j$, donde η es el número de diferentes tipos de borde (y también el número de matrices correspondientes).

6.3.3 Esqueletización y análisis estructural

Las técnicas de extracción de características discutidas en los capítulos anteriores están basadas en el procesamiento estadístico de la imagen. Estos dos métodos no consideran los aspectos estructurales de las imágenes analizadas. Las pequeñas diferencias en los mapas de bit, a veces significan una gran diferencia en la estructura de los caracteres contenidos. Por ejemplo, los dígitos 6 y 8 tienen un mapa de bit similar, aunque hay una sustancial diferencia en sus estructuras.

El análisis estructural está basado en conceptos mayores que el método de detección de bordes. Esto hace no tratar con términos como “pixel” o “bordes”, pero se consideran estructuras más complejas (como uniones, extremos de líneas o bucles). Para analizar estas estructuras, debemos introducir el un algoritmo de adelgazamiento para obtener el esqueleto del carácter. Este capítulo se ocupa del principio de esqueletización, así como los principios del análisis estructural de la imagen obtenida durante el proceso.

6.3.3.1 Concepto de esqueletización

La esqueletización es una reducción de la forma estructural de una determinada figura. Esta reducción se consigue mediante la obtención de un esqueleto de la región a través del algoritmo de esqueletización. El esqueleto de una forma se define matemáticamente como una transformación del eje medio. Para definir la transformación del eje medio y el algoritmo de esqueletización, debemos introducir algunos requisitos previos.

Sea \check{N} una relación binaria entre dos pixeles $[x, y]$ y $[x', y']$, tal que $a\check{N}b$ significa que “ a es vecino de b ” (véase Figura 38. Ejemplo de conexión de 4 pixeles u 8 pixeles.). Esta relación se define como:

$$[x, y]\check{N}_8[x', y'] \leftrightarrow |x - x'| = 1 \cup |y - y'| = 1 \text{ para vecindades de 8 píxeles}$$

$$[x, y]\check{N}_4[x', y'] \leftrightarrow |x - x'| = 1 \oplus |y - y'| = 1 \text{ para vecindades de 4 píxeles}$$

El borde B del carácter es un conjunto de pixeles límite. El pixel $[x, y]$ es un *pixel límite*. Si éste es negro y tiene al menos un pixel vecino de color blanco en la vecindad de 8 pixeles:

$$[x, y] \in B \leftrightarrow f(x, y) = 0 \cap \exists [x', y']: f(x', y') = 1 \cap [x, y]\check{N}_8[x', y']$$

La región interior I del carácter es un conjunto de píxeles negros, que no son píxeles frontera:

$$[x, y] \in I \leftrightarrow f(x, y) = 0 \cap [x, y] \notin B$$

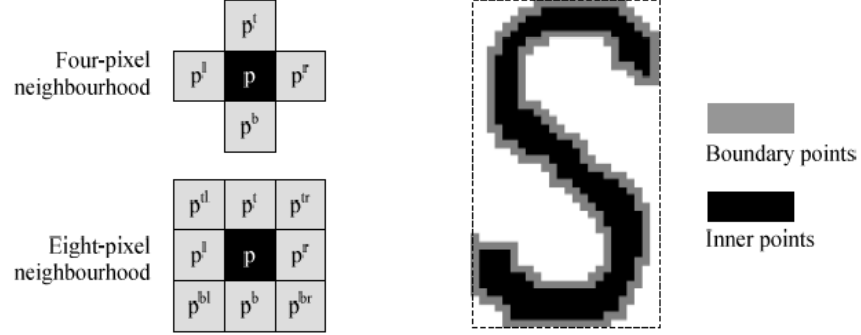


Figura 48. Conjunto de píxeles interiores y límite del carácter.

La pieza P es entonces una unión de todos los píxeles límite e interiores ($P = B \cup I$). Dado que solo hay un solo grupo continuo de píxeles negros, todos estos pertenecen a la pieza R . El principio y la terminología de la esqueletización son similares al algoritmo de extracción de piezas o partes, desarrollado en apartados anteriores.

6.3.3.2 Transformación del eje medio

La transformación del eje medio de la pieza P se define de la siguiente manera. Para cada píxel interior $p \in I$, encontramos el píxel vecino más cercano $p_b \in B$. Si un píxel p tiene más de un vecino, se dice que pertenece al eje medio (o esqueleto) de la pieza o parte P . El concepto del píxel vecino más cercano depende de la definición de la distancia euclídea entre dos píxeles en el sistema ortogonal de coordenadas. Matemáticamente, el eje medio S (o esqueleto) es un subconjunto de R definido como:

$$p \in S \leftrightarrow \exists p_1 \exists p_2: p_1 \in B \cap p_2 \in B \cap d(p, p_1) = \min_{p' \in B} \{d(p, p')\}$$

El píxel p pertenece al eje medio S , si existen al menos dos píxeles p_1 y p_2 , tal que la distancia euclídea entre los píxeles p y p_1 es igual a la distancia entre los píxeles p y p_2 , y estos mismos píxeles son los límites más cercanos al píxel p . La distancia euclídea entre dos píxeles $p_1 = [x_1, y_1]$ y $p_2 = [x_2, y_2]$ se define como:

$$d(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

6.3.3.3 Algoritmo de esqueletización

La implementación de la definición matemática de la transformación del eje medio o medial es computacionalmente costosa, ya que se trata de calcular la distancia de cada pixel interno del conjunto I con cada pixel en los límites de B .

La transformación del eje medio está intuitivamente definida por un concepto llamado “frente de fuego o fire front”. Se considera retoricamente que “un fuego prende a lo largo del límite”. Todos los frentes de ese “supuesto fuego” avanzan dentro del interior del carácter a la misma velocidad. El esqueleto de un carácter es entonces un conjunto de pixeles alcanzados por más de un frente de fuego al mismo tiempo.

El algoritmo de esqueletización (o adelgazamiento) se basa en el concepto de frente de fuego o fire front. El adelgazamiento es una operación morfológica, que conserva finales de píxeles y no rompe la conectividad. Suponemos que los píxeles de la pieza son negros (valor 0) y los pixeles de fondo blancos (valor 1).

El adelgazamiento es un proceso iterativo formado por dos pasos sucesivos aplicados a los píxeles límite o frontera de una pieza. Con referencia a la notación de vecindad de 8 pixeles, el primer paso señala un pixel p límite para su eliminación, si cada una de las siguientes condiciones se satisfacen:

- Al menos un vecino superior o inferior derecho del pixel p debe ser blanco (el pixel p es blanco cuando éste no pertenece a la pieza P).

$$p^t \notin P \cup p^r \notin P \cup p^b \notin P$$

- Al menos un vecino superior o inferior izquierdo del pixel p debe ser blanco.

$$p^l \notin P \cup p^r \notin P \cup p^b \notin P$$

- El pixel p debe tener al menos dos, y más de seis vecinos de color negro de la pieza R . Esta condición impide al algoritmo el borrado de puntos finales y ruptura de la conectividad.

$$2 \leq |\{p' | p \check{N}_8 p' \cap p' \notin P\}| \leq 6$$

- El número de transiciones de blanco a negro en la secuencia ordenada $p^t, p^{tr}, p^r, p^{br}, p^b, p^{bl}, p^l, p^{tl}, p^t$ debe ser igual a uno.

$$v(p^t \in P \cap p^{tr} \notin P) + v(p^{tr} \in P \cap p^r \notin P) + v(p^r \in P \cap p^{br} \notin P) + v(p^{br} \in P \cap p^b \notin P) + v(p^b \in P \cap p^{bl} \notin P) + v(p^{bl} \in P \cap p^l \notin P) + v(p^l \in P \cap p^{tl} \notin P) + v(p^{tl} \in P \cap p^t \notin P) = 1$$

$$v(x) = \begin{cases} 0 & \text{si } x \\ 1 & \text{si } \text{No } x \end{cases}$$

El primer paso marca el pixel p para su eliminación, si sus vecindades cumplen con las condiciones anteriores. Sin embargo, el pixel no es eliminado hasta que todos los otros pixeles han sido procesados. Si al menos no se satisface una de las condiciones, el valor del pixel p , no se cambia.

Después de que el paso primero ha sido aplicado a todos los pixeles vecinos, los pixeles marcados son definitivamente borrados en la segunda etapa. Cada iteración de estos dos adelgaza el carácter procesado. Este proceso iterativo se aplica hasta que ya no existan pixeles marcados para su borrado. El resultado del algoritmo de adelgazamiento es un esqueleto (o eje medio) del carácter procesado.

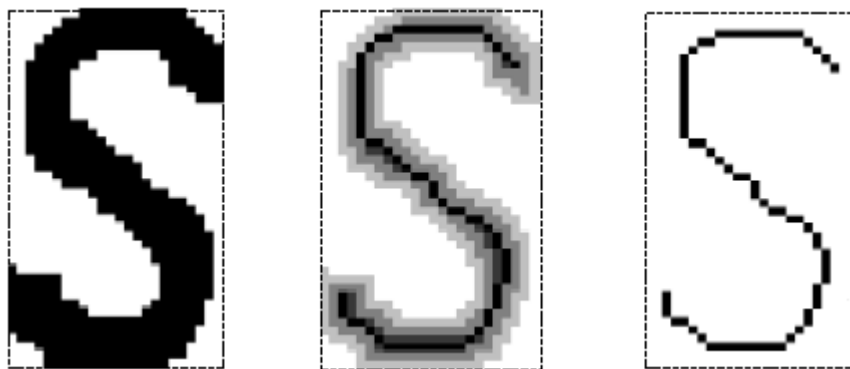


Figura 49. Mapa de bits del carácter durante el proceso de esqueletización.

Como observamos en la figura anterior el algoritmo de adelgazamiento elimina iterativamente los píxeles vecinos. Los pixeles eliminados en la primera iteración están marcados por un color gris pastel. Los pixeles eliminados en la segunda y tercera iteración están marcado por un gris más oscuro. El resultado del algoritmo de adelgazamiento es un “esqueleto del carácter” (o eje medio).

6.3.3.4 Análisis estructural del resultado

El análisis estructural es un método de extracción de características que considera estructuras más complejas que los simples píxeles. La idea básica es que la diferencia sustancial entre dos caracteres comparados no puede ser evaluada por un análisis estadístico. Debido a esto, el análisis estructural extrae características, que no describen pixeles o bordes, pero sí estructuras más complejas, tales como uniones, extremos de línea y bucles.

6.3.3.5 Unión

La unión es un punto, que tiene al menos tres vecinos de color negro en una vecindad de 8 pixeles. Consideramos solo dos tipos de uniones: la unión de 3 y 4 líneas. El número de uniones en la procesada pieza R se define matemáticamente como:

$$n_j^3 = |\{p | \exists^3 p' : \{p, p'\} \subseteq P \cap p\ddot{N}_8 p'\}|$$

$$n_j^4 = |\{p | \exists^4 p' : \{p, p'\} \subseteq P \cap p\ddot{N}_8 p'\}|$$

6.3.3.6 Fines de línea

El fin de línea es un punto, que tiene exactamente un vecino en una vecindad de 8 pixeles. El número de fines de líneas en una pieza P se define como:

$$n_e = |\{p | \exists! p_1 : \{p, p_1\} \subseteq P \cap p\ddot{N}_8 p_1\}|$$

6.3.3.7 Bucles

No es fácil determinar el número de bucles n_l en el carácter modificado. El algoritmo está basado en el siguiente principio. Primeramente, debemos complementar el mapa de bits del carácter adelgazado. Los pixeles negros serán considerados como fondo y los pixeles blancos como primer plano. El número de bucles en la imagen es igual al número de “lagos”, que son rodeados por estos bucles. Puesto que ese “lago” es un grupo continuo de pixeles blancos en la imagen positiva, se aplica el algoritmo de extracción de piezas en la imagen negativa para determinan el número de piezas de color negro. Por tanto, el número de bucles es igual al número de piezas de color negro menos uno, ya que una pieza representa el fondo de la imagen original (complementada o negada es el primer plano). Otra manera sería utilizar una serie de erosiones morfológicas.

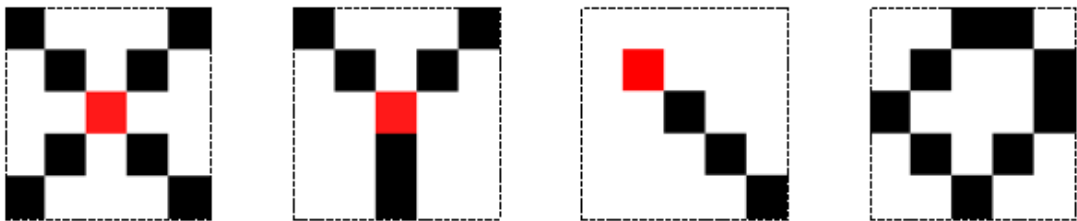


Figura 50. Ejemplos de unión, fin de línea y bucle respectivamente.

Dado que no sabemos el número de bordes del esqueleto, no podemos usar una ecuación ciclomática estándar que sabemos de la teoría de las gráficas. Además, dos esqueletos similares del mismo carácter a veces pueden diferir en un número de uniones. Debido a esto, no se recomienda usar limitaciones basadas en el número de uniones.

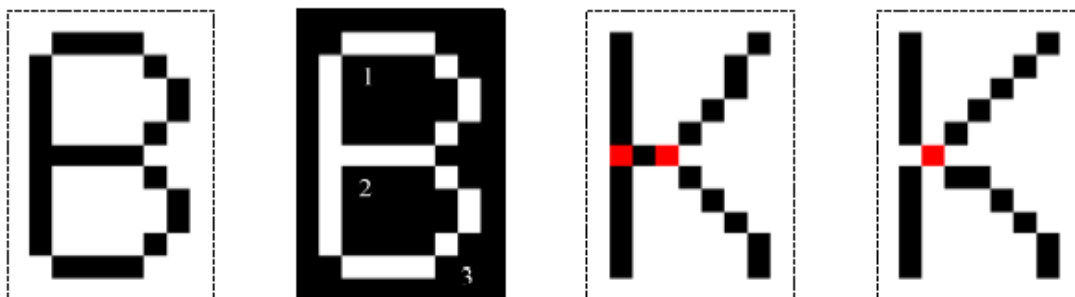


Figura 51. Ilustración y proceso del término “lago” o “lazo”.

Como vemos en la Figura 51. Ilustración y proceso del término “lago”, determinamos el número de lagos en el esqueleto del carácter aplicando el algoritmo de extracción de piezas en la imagen negativa. La imagen negativa contiene tres piezas. Dado que la pieza 3 es un fondo, solo dos piezas están consideradas como lagos en la imagen. En las dos siguientes representaciones, podemos ver como esqueletos similares del mismo carácter pueden diferir en el número de uniones

6.3.3.8 Limitaciones estructurales

Para mejorar el proceso de reconocimiento, se puede asumir las limitaciones estructurales mostradas en la tabla 3. El análisis sintáctico se puede combinar con otros métodos descritos en capítulos anteriores, tales como el *método de detección de bordes* o la *matriz de píxeles*.

La forma más sencilla es usar una red neural global, que devuelva varios candidatos y entonces seleccione el mejor candidato, es que cumpla las limitaciones estructurales. La solución más sofisticada es el uso de limitaciones estructurales para la selección adaptativa de la red neuronal local.

| | Fines de línea | Bucles | Uniones |
|---|----------------------|-----------------------------|----------------------|
| 0 | BDO08 | CEFGHIJKLMNOPSTUVWXYZ123457 | CDGIJLMNOSUVWZ012357 |
| 1 | PQ69 | ADOPQR09 | EFKPQTX469 |
| 2 | ACGIJLMNRSUVWZ123457 | B8 | ABHR8 |
| 3 | EFTY | | |
| 4 | HKX | | |

Tabla 3. Limitaciones estructurales de todos los caracteres que pueden ser utilizados.

Las limitaciones estructurales pueden ser aplicadas antes o después del reconocimiento por redes neuronales.

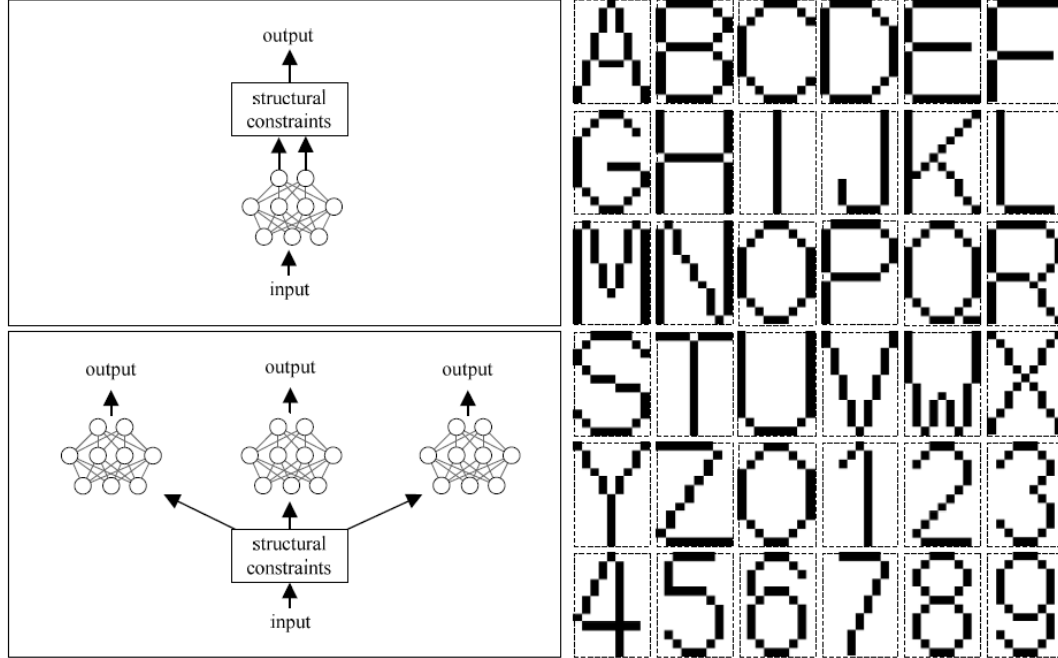


Figura 52. (a) Invariabilidad anterior o posterior de la aplicación de las limitaciones estructurales y (b) ejemplo de alfabeto después de la esquelización.

6.3.3.9 Extracción de características

En caso de que sepamos la posición de las posiciones estructurales, se puede formar un vector de descriptores directamente a partir de esta información. Supongamos que hay varios fines de línea, bucles y uniones en la imagen. La posición del bucle se define por su centro. Para formar el vector debemos convertir las coordenadas estructurales del elemento en coordenadas polares $[r, \theta]$:

$$r = \sqrt{x'^2 + y'^2} ; \theta = \arctan\left(\frac{y'}{x'}\right) ; x' = \frac{2 \cdot x - w}{w} ; y' = \frac{2 \cdot y - h}{h}$$

donde x' e y' son las coordenadas rectangulares normalizadas.

La longitud y la estructura del vector resultante variará de acuerdo a un número y tipo de elementos estructurales contenidos en el carácter. Dado que las limitaciones estructurales dividen los caracteres en varias clases, hay varios

tipos posibles de *vector de descripción*. Cada tipo de vector se corresponde con una clase de carácter.

Por ejemplo, sea un carácter con dos fines de línea y una unión. Esta limitación determina la siguiente clase de tipos de caracteres posibles: (G, I, J, L, M, N, S, U, V, W, Z, 1, 2, 3, 5, 7). Definimos un vector de descriptores para distinguir entre estos caracteres de la siguiente manera:

$$x = (r_1, \theta_1, r_2, \theta_2, r_3, \theta_3)$$

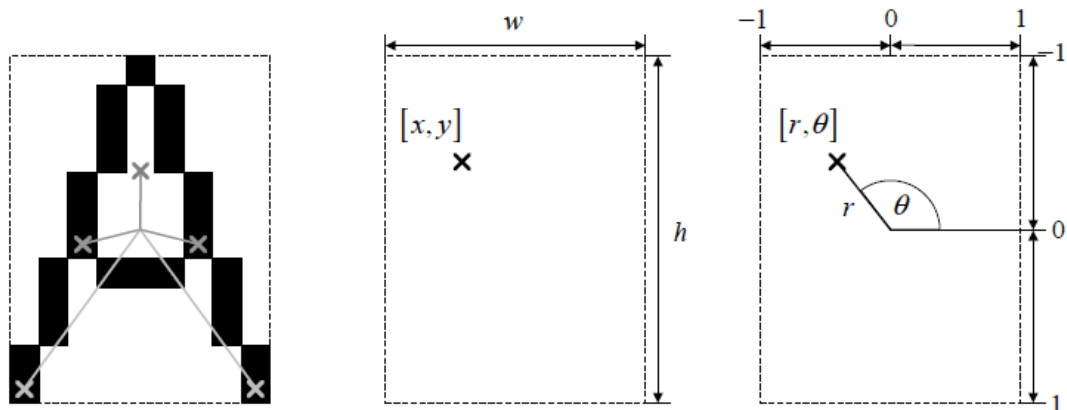


Figura 53. Ilustración del proceso de extracción de características.

En la ilustración anterior podemos ver un ejemplo, en el que el esqueleto del carácter contiene varios elementos estructurales, como uniones, bucles y fines de línea. Cada elemento puede estar posicionado en el sistema de coordenadas rectangular o polar.

7 Reconocimien to óptico de caracteres OCR

En el capítulo anterior hemos explicado los distintos métodos de extracción de características. El objetivo de estos métodos es el de obtener un vector de descriptores (denominado patrón), que de forma integral describe el carácter contenido en el mapa de bits procesado. El objetivo de este capítulo es introducir las técnicas de reconocimiento de patrones, tales como redes neuronales, que son capaces de clasificar los patrones en las clases correspondientes.

7.1 Clasificación del problema general

El problema de la clasificación general se formula mediante la asignación entre los elementos de dos conjuntos. Sea **A** un conjunto de todas las posibles combinaciones de descriptores, y **B** un conjunto de todas las clases. La clasificación se entiende como la proyección del grupo de elemento similar desde el conjunto **A** dentro de una clase común representada por un elemento en el conjunto **B**. Así un elemento en el conjunto **B** corresponde a una clase. Normalmente el grupo de casos distinguibles del mismo carácter se corresponde a una clase, pero en ocasiones una clase representa dos caracteres mutuamente indistinguibles entre sí, como “0” y “O”.

Sea F una hipotética función que asigna cada elemento desde el conjunto **A** a un elemento del conjunto **B**;

$$F: A \rightarrow B$$

$$\hat{x} = F(x)$$

donde $x \in A$ es un vector de descripción (patrón) que describe la estructura del carácter clasificado y $\hat{x} \in B$ es un clasificador, que representa la semántica de dicho carácter.

La función F es el clasificador teórico probablemente mejor, pero su construcción es imposible, dado que no podemos tratar con cada combinación de descriptores. En la práctica, se construye un clasificador de parámetros usando solamente un subconjunto limitado de las asignaciones de $A \rightarrow B$. Este subconjunto se conoce como un “training set” o “conjunto de entrenamiento”, tal que $A_t \subset A$ y $B_t \subset B$. Nuestro objetivo es construir una aproximación $\tilde{F}(x, w)$ de la hipotética función F , donde w es un parámetro que afecta a la calidad de la aproximación:

$$\tilde{F}(w): A_t \rightarrow B_t$$

$$\hat{x} = \tilde{F}(x, w)$$

donde $x \in A_t \subset A$, $\hat{x} \in B_t \subset B$. Formalmente, podemos decir que $\tilde{F}(w)$ es una restricción de la proyección F sobre un conjunto $A_t \subset A$. Asumimos que para cada $x_i \in A_t$ sabemos el valor deseado $\hat{x}_i \in B_t$:

$$x_0 \rightarrow \hat{x}_0, x_1 \rightarrow \hat{x}_1, x_2 \rightarrow \hat{x}_2, \dots, x_{n-1} \rightarrow \hat{x}_{n-1}$$

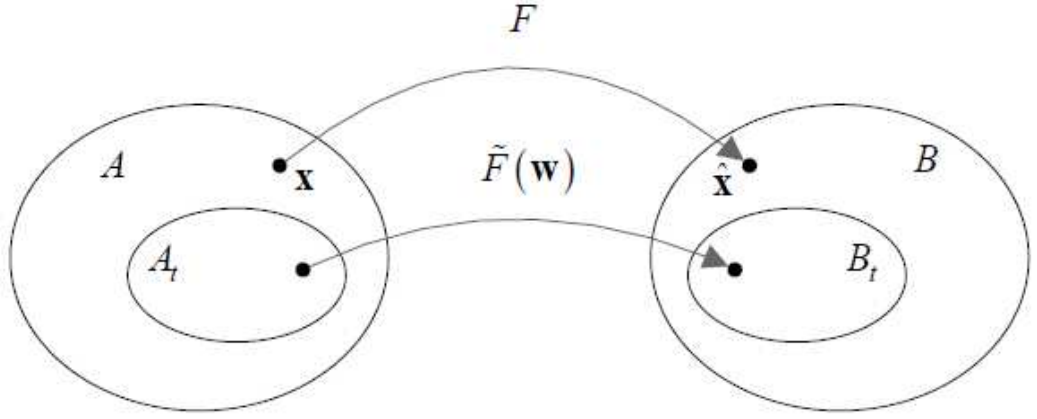


Figura 54. Proyección entre los conjuntos A y B.

En el ejemplo, vemos que F es una función hipotética que asigna cada posible combinación de un patrón de entrada $x \in A$ a una correspondiente clase de $x \in B$. Esta proyección está aproximada a una función $\tilde{F}(w)$ que traza patrones de entrada del training set A_t dentro de las clases correspondientes del conjunto B_t .

El problema es encontrar un valor o valores óptimos para el parámetro w . La w es típicamente un vector (o matriz) de pesos sintácticos en una red neuronal. De acuerdo con este parámetro, los valores de la función $\tilde{F}(x, w)$ debería ser lo más próxima posible de los valores de $F(x)$ para los patrones de entrada x del training set A_t . Definimos una función de error para evaluar la solvencia del parámetro w :

$$E(w) = \frac{1}{2} \sum_{i=0}^{m-1} (\tilde{F}(x_i, w) - \hat{x}_i)^2$$

donde m es el número de patrones $x_0 \dots x_{m-1}$ en el training set A_t . Sea w_+ un valor óptimo del parámetro w , tal que $w_+ = \arg \min_{w \in W} \{E(w)\}$. Por tanto la aproximación $\tilde{F}(x, w_+)$ de la función $F(x)$ se considera como adaptada. La aproximación adaptada $\tilde{F}(x, w_+)$ simula la función original $F(x)$ para los

patrones x del training set A_t . Además, esta aproximación es capaz de predecir el clasificador de salida \hat{x} para patrones desconocidos x del conjunto de test A_x ($A_x = A - A_t$). La función con tal capacidad de predicción sustituye parcialmente el hipotético clasificador $F(x)$. Dado que la función $F(x, w)$ es solo un modelo, usamos una red neuronal feed-forward (en español “prealimentada”) para esta implementación.

7.2 La neurona biológica y sus modelos matemáticos

Para una mejor comprensión de la arquitectura de una red neuronal artificial, es necesario explicar la estructura y la funcionalidad de una neurona biológica.

El cerebro humano es una red neuronal de aproximadamente diez billones de neuronas interconectadas. Cada neurona es una célula que utiliza una reacción bioquímica para procesar y transmitir información. Una célula nerviosa tiene un cuerpo del tamaño de varias micras y miles de conexiones de entrada llamadas “dendritas”. También cuenta con una conexión de salida llamada “axón”, que puede ser de varios metros de largo. El flujo de datos en la red neuronal biológica es representada por señal eléctrica, que se propaga a lo largo del axón. Cuando la señal llega a una conexión sináptica entre un axón y una dendrita consecutiva, ésta libera moléculas de agente químico (llamado mediadores o neurotransmisores) dentro de tal dendrita. Esta acción hace que se produzca un cambio local de polaridad en la membrana de transmisión de la dendrita. La diferencia en la polaridad de la membrana de transmisión activa una ola de potencial dendrita-somática, que avanza en un sistema de dendritas ramificadas dentro del cuerpo de una neurona.

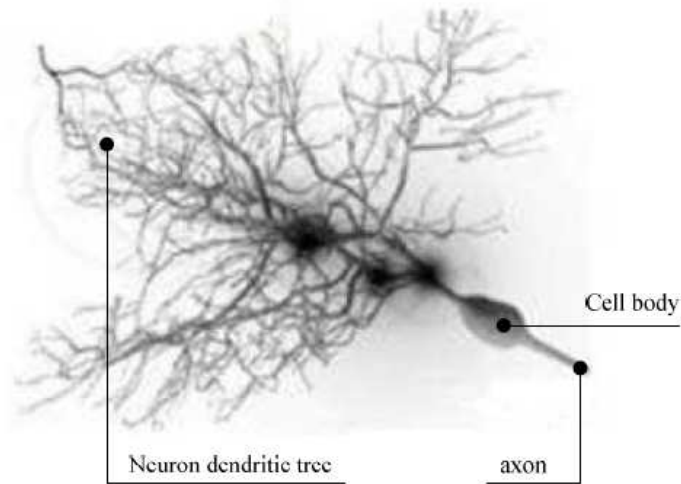


Figura 55. La neurona biológica.

La red neuronal biológica contiene dos tipos de conexiones sinápticas. La primera es una conexión excitadora, que amplifica la señal. La segunda (inhibidora) conexión suprime la señal. El comportamiento de la conexión está representada por su “peso”. La red neuronal contiene un mecanismo que es capaz de alterar los pesos de las conexiones. Por ello, el sistema de pesos sinápticos es la base de la construcción de la memoria humana. Como los pesos son continuamente alterados, la información antigua se va olvidando poco a poco.

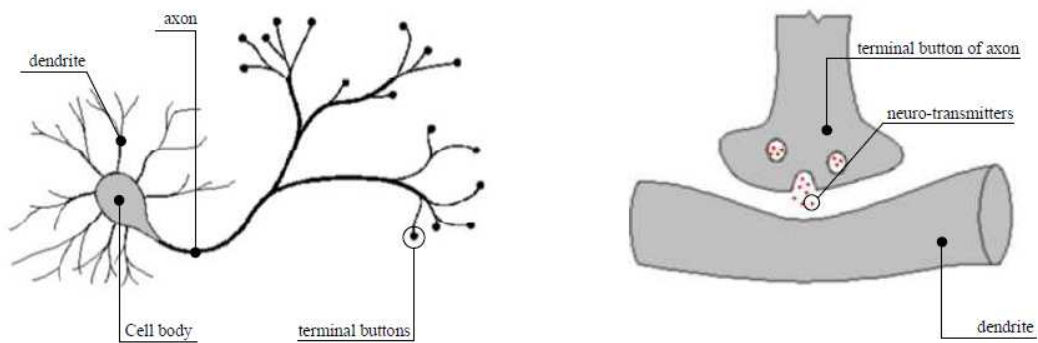


Figura 56. (a) Ilustración esquemática de una célula neuronal y (b) Conexión sináptica entre una dendrita y un botón terminal del axón.

Dado que la problemática de la neurona biológica es muy compleja, los científicos propusieron varios métodos matemáticos, como el de la *neurona de McCulloch-Pitts* y el *perceptrón* (o *perceptron* en inglés), por lo que a partir de ahora, en los párrafos siguientes siempre nos referiremos a dicho tipo de neurona.

7.2.1 La neurona de McCulloch-Pitts

La neurona de McCulloch-Pitts fue el primer modelo de neurona matemática propuesto. Esta neurona solo tiene dos posibles valores de salida (0 o 1) y solo dos tipos de pesos sinápticos: el completamente excitante y el completamente inhibidor. El peso excitante (1) no afecta a la entrada, pero el inhibidor produce una negación (-1 al valor de entrada). Los pesos de entrada son contados y procesados conjuntamente por una neurona de la siguiente manera:

$$y = g \left(\sum_{j=0}^{J-1} w_{i,j} \cdot x_j - \vartheta_i \right)$$

$$g(\xi) = \begin{cases} 0 & \text{si } \xi < 0 \\ 1 & \text{si } \xi \geq 0 \end{cases}$$

Este tipo de neurona puede realizar funciones lógicas como AND, OR y NOT. Además McCulloch y Pitts demostraron que una matriz síncrona de tales neuronas es capaz de realizar una función arbitraria computacional, de una manera similar a la máquina de Turing. Dado que las neuronas biológicas no tienen respuesta binaria (pero sí continua), este modelo de neurona no es apropiado para nuestra aproximación.

7.2.2 Perceptrón

Otro modelo de neurona es el “perceptron” o “perceptrón”. Se ha demostrado que las redes McCulloch-Pitts con conexiones sinápticas modificadas pueden ser entrenadas para el reconocimiento y clasificación. Dicho entrenamiento está basado en una modificación de los pesos neuronales, de acuerdo a la reacción de dichas neuronas. Si la neurona no es activa y debería estarlo, incrementamos sus pesos. Si la neurona está activa y debería no estarlo, los disminuimos. Este principio ha sido utilizado en un primer modelo de clasificador neuronal llamado ADALINE (adaptive linear neuron). El problema principal de tales redes es que no son capaces de resolver problemas linealmente no separables.

Este problema fue resuelto cuando los científicos, Rumelhart, Hilton y Williams que propusieron el método de aprendizaje del error de propagación inversa para redes perceptrón multicapa. Las sencillas neuronas matemáticas de McCulloch-Pitts han sido reemplazadas por neuronas con una función de saturación continua de entrada/salida.

El perceptrón tiene múltiples entradas análogas (dendritas) y una sola salida (correspondiente a un axón). Sea $x_0 \dots x_{j-1}$ parámetros de entrada con sus correspondientes ponderaciones $w_{i,0} \dots w_{i,j-1}$. Las entradas ponderadas están contadas, umbralizadas y saturadas conjuntamente de la siguiente manera:

$$y = g \left(\sum_{j=0}^{J-1} w_{i,j} \cdot x_j - \vartheta_i \right)$$

$$g(\xi) = \frac{1}{1 + e^{-\xi}}$$

donde $g(\xi)$ es una función de saturación sigmoidea y ϑ_i es el valor de umbral. En ocasiones, el umbral se implementa como un valor de entrada con una constante ponderada de -1. Por tanto, la función de una neurona puede ser simplificada a $y = g(x \cdot w)$, donde x es un vector con valores de entrada (incluyendo el valor umbral), y w es un vector de pesos (incluyendo la constante de ponderación -1).

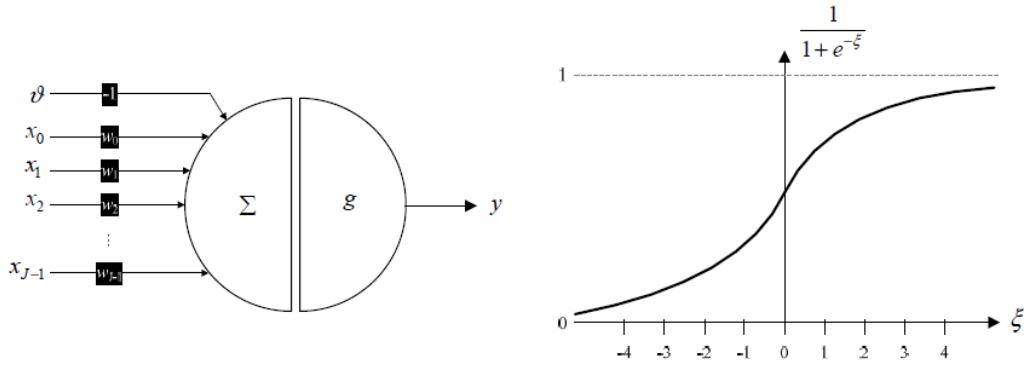


Figura 57. (a) Función del sumatorio y la ganancia (saturación) g del perceptrón con un umbral implementado como un valor de entrada dedicado. (b) Función de saturación sigmoidea.

7.3 Red neuronal prealimentada

De manera formal, la red neuronal se define como un gráfico orientado $G = (N, E)$, donde N es un conjunto neuronas matemáticas no vacío, y E es un conjunto de conexiones orientadas entre neuronas. La conexión $e(n, n') \in E$ es una relación binaria entre dos neuronas n y n' . El conjunto de todas las neuronas N está compuesto por los conjuntos disyuntivos N_0, N_1, N_2 , donde N_i es un conjunto de todas las neuronas de la i -enésima capa.

$$N = N_0 \cup N_1 \cup N_2$$

El peso j de una neurona i en una capa k se denota como $w_{i,j}^{(k)}$ y el umbral de una neurona i en una capa k se denota como $\vartheta_i^{(k)}$. Los números de neuronas para la capa de entrada (0), oculta(1) y salida (2) se denotan como m, n, o , tal que $m = |N_0|$, $n = |N_1|$, y $o = |N_2|$.

El número de neuronas en la capa de entrada (m) es igual a la longitud de un patrón de entrada x , para que cada valor del patrón esté dedicado a una neurona. Las neuronas en la capa de entrada no realizan alguna función de cálculo, si no que solo distribuyen los valores de un patrón de entrada a las neuronas en la capa oculta. Debido a esto, la neurona de la capa de entrada tiene una entrada directamente asignada en múltiples salidas. Por ello, el valor de umbral $\vartheta_i^{(0)}$ de la capa de entrada es igual a cero, y los pesos de entrada $w_{i,j}^{(0)}$ son igual a 1.

El número de neuronas en la capa oculta (n) es escalable, pero esto afecta a las habilidades de reconocimiento de la red neuronal en su conjunto. Muy pocas neuronas en la capa oculta hacen que la red neuronal no sean capaces de aprender nuevos patrones. Demasiadas neuronas causan que la red se sobreexplote, por lo que no sería capaz de generalizar patrones desconocidos.

La información en una red neuronal prealimentada se propaga desde capas inferiores a las superiores por conexiones unidireccionales. Hay conexiones entre capas solamente adyacentes, de este modo no contienen conexiones retroalimentadas (de capas superiores a inferiores) o conexiones entre dos capas arbitrarias. Además no existen conexiones entre neuronas de la misma capa.

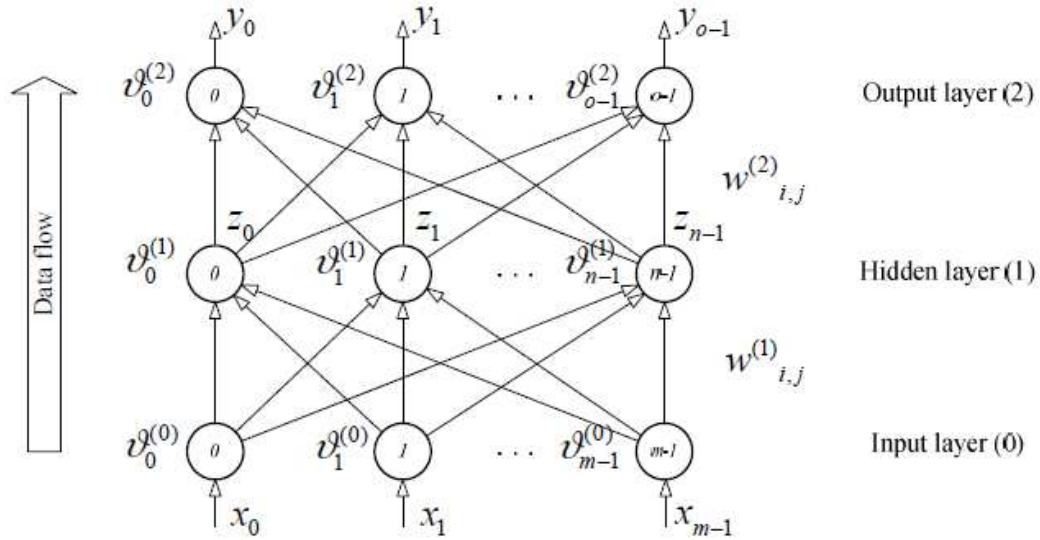


Figura 58. Arquitectura de una red neuronal prealimentada de tres capas.

7.4 Adaptación al mecanismo de red neuronal prealimentado

Se ha demostrado que una red neural multicapa compuesta de perceptrones con una función de saturación sigmoide, puede solucionar un problema arbitrario no lineal. Matemáticamente para cada función $F: \mathbb{R}^m \rightarrow \mathbb{R}^o$, en las cuales existe una red neuronal prealimentada multicapa, son capaces de realizar esta función. La prueba está basada en el *teorema de Kolmogorov*, la cual dice que cada función continua y creciente f definida en el intervalo $\langle 0,1 \rangle^m$ puede ser escrita como:

$$f(x_0 \dots x_{m-1}) = \sum_{i=0}^{2 \cdot m} \alpha_i \left(\sum_{j=0}^{m-1} \phi_{i,j}(x_j) \right)$$

donde α_i son funciones continuas con un parámetro apropiadamente escogido.

El problema se centra en cómo construir una red neuronal correspondiente a una función no lineal dada. Primeramente, se elige una topología propia de la red. El número de neuronas en la capa de entrada y de salida viene dada por la longitud de los patrones de entrada y de salida, mientras que el número de neuronas de la capa oculta es escalable.

Una adaptación de la red neural o neuronal significa encontrar el parámetro óptimo w_+ de la función de aproximación $\tilde{F}(x, w)$ discutida anteriormente. Sean definidas dos funciones de error para evaluar la utilidad del parámetro w :

$$E_t = \frac{1}{2} \sum_i^{A_t} (\tilde{F}(x_i, w) - \tilde{x}_i)^2 ; E_x = \frac{1}{2} \sum_i^{A_x} (\tilde{F}(x_i, w) - \tilde{x}_i)^2$$

donde el subíndice “t” significa “training”(entrenamiento), y “x” significa “test”. E_t es una función de error definida para patrones del training set, y E_x para patrones del conjunto de test. La respuesta de la red neural a un patrón d entrada x es dado como $y_i = \tilde{F}(x_i, w)$.

La función de error E_t decrece a medida que el número de neuronas en la capa oculta aumenta. Esta relación es también válida entre la función E_t y un número de pasos iterativos del proceso de adaptación. Estas relaciones pueden ser matemáticamente descritas de la siguiente manera:

$$\lim_{n \rightarrow \infty} E_t = 0 ; \lim_{k \rightarrow \infty} E_t = 0$$

Donde n es el número de neuronas en la capa de entrada y k es el número de pasos iterativos del proceso de adaptación.

La función de error E_x no tiene un límite en cero como n y k , sino que tiende a infinito. Debido a esto , existe un número óptimo de neuronas y otro de pasos iterativos, en los cuales la función E_x tiene un mínimo.

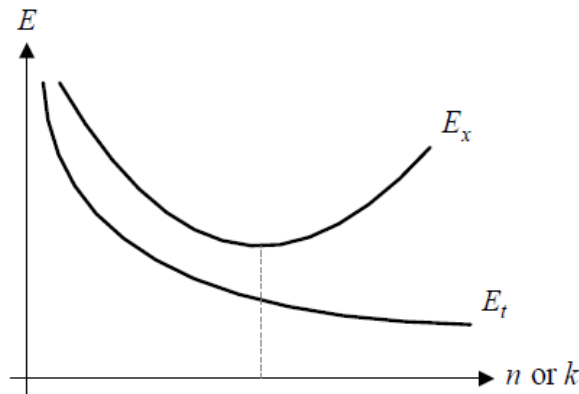


Figura 59. Dependencia de las funciones de error E_t y E_x en el número de neuronas en la capa de entrada (n) y en el número de pasos iterativos (k).

Por simplicidad, vamos a suponer solo una red neural prealimentada con una sola capa de neuronas ocultas. Todas las neuronas en las capsas adyacentes

están conectadas por conexiones orientadas. No existen conexiones retroalimentadas, o conexiones entre neuronas dentro de una sola capa. Las actividades de las neuronas ocultas y de salida se definen como:

$$z_i = g \left(\sum_{j=0}^{m-1} w_{i,j}^{(1)} \cdot x_j - \vartheta_i^{(1)} \right); y_i = g \left(\sum_{j=0}^{n-1} w_{i,j}^{(2)} \cdot z_j - \vartheta_i^{(2)} \right)$$

donde $g(\xi)$ es una función de saturación sigmoidea.

7.4.1 Fase activa

La evaluación de las actividades de las neuronas ocultas y de salida se realiza en la llamada “fase activa”. La *fase activa* consiste en dos pasos de redes neuronales de tres capas. El primer paso es una evaluación de actividades z_i en la capa oculta, y el segundo es una evaluación de las actividades en y_i . Dado que la evaluación de las actividades se realiza desde las capas inferiores a las superiores, el término “feed-forward o alimentación progresiva o prealimentación” se refiere este principio, y se realiza cada vez que hay la necesidad de clasificar el parámetro de entrada x .

7.4.2 Derivadas parciales y gradiente de la función de error

El objetivo de la fase de entrenamiento es encontrar valores óptimos de umbrales y ponderaciones para minimizar la función de error E_t . La fase de adaptación es un proceso iterativo en el que una respuesta y a un patrón de entrada x es comparado con la respuesta deseada \hat{x} . La diferencias entre la respuesta deseada y obtenida se usa para una corrección de pesos. Los pesos son alterados iterativamente hasta que el valor de la función de error E_t es insignificante.

7.4.2.1 Gradiente de la función de error afín con un patrón único

Calculamos un gradiente g de una función de error afín a un patrón único x deseado, y obtenemos los resultados y y \hat{x} . El gradiente g se calcula en la dirección de las capas superiores hacia las inferiores como se muestra a continuación.

Primero, calculamos los componentes del gradiente afines a los umbrales $\vartheta_i^{(2)}$ en la capa de salida como:

$$\frac{\partial E}{\partial \vartheta_i^{(2)}} = (y_i - \hat{x}_i) \cdot (1 - y_i) \cdot y_i$$

Entonces, se calculan los componentes del gradiente de acuerdo con los umbrales $\vartheta_i^{(1)}$ en la capa oculta. Éstos se calculan a partir de los componentes $\frac{\partial E}{\partial \vartheta_i^{(2)}}$ del paso anterior de la siguiente manera:

$$\frac{\partial E}{\partial \vartheta_i^{(1)}} = z_i \cdot (1 - y_i) \cdot \sum_{j=0}^{o-1} \frac{\partial E}{\partial \vartheta_i^{(2)}} w_{i,j}^{(2)}$$

De manera similar, calculamos los componente del gradiente relacionados con los pesos $w_{i,j}^{(2)}$ y $w_{i,j}^{(1)}$:

$$\frac{\partial E}{\partial w_{i,j}^{(2)}} = \frac{\partial E}{\partial \vartheta_i^{(2)}} \cdot z_j, \quad \frac{\partial E}{\partial w_{i,j}^{(1)}} = \frac{\partial E}{\partial \vartheta_i^{(1)}} \cdot x_j$$

El gradiente g es un vector de componentes obtenido de la siguiente manera:

$$g = \frac{\partial E}{\partial \vartheta_0^{(1)}}, \dots, \frac{\partial E}{\partial \vartheta_{n-1}^{(1)}}, \frac{\partial E}{\partial \vartheta_0^{(2)}}, \dots, \frac{\partial E}{\partial \vartheta_{o-1}^{(2)}}, \frac{\partial E}{\partial w_{0,0}^{(1)}}, \dots, \frac{\partial E}{\partial \vartheta_{n-1,m-1}^{(1)}}, \frac{\partial E}{\partial w_{0,0}^{(2)}}, \dots, \frac{\partial E}{\partial \vartheta_{o-1,n-1}^{(2)}}$$

7.4.2.2 Gradiente global

El gradiente general o global se define como una suma de los gradientes relacionados con los patrones del training set A_t . Sea $g_{x/\hat{x}}$ un gradiente afín con un par de formación x/\hat{x} , el gradiente general se calcula como:

$$\sum_{x/\hat{x}}^{A_t} g_{x/\hat{x}}$$

7.4.3 Fase de adaptación

La fase de adaptación es un proceso iterativo de búsqueda óptima de valores de peso y umbral, para los cuales un valor de error de la función E_t está en un mínimo local. La Figura 60. Aproximación numérica de la búsqueda del mínimo global en el "landscape error". muestra esquemáticamente una gráfica de la función E_t llamada "landscape error" (o "error de paisaje" en su traducción). Generalmente, el "landscape error" es $|w| + 1$ dimensional, donde $|w|$ es cardinalidad del vector de umbrales y pesos, tal que:

$$w = \left(\vartheta_0^{(1)}, \dots, \vartheta_{n-1}^{(1)}, \vartheta_0^{(2)}, \dots, \vartheta_{o-1}^{(2)}, w_{0,0}^{(1)}, \dots, w_{n-1,m-1}^{(1)}, w_{0,0}^{(2)}, \dots, w_{o-1,n-1}^{(2)} \right)$$

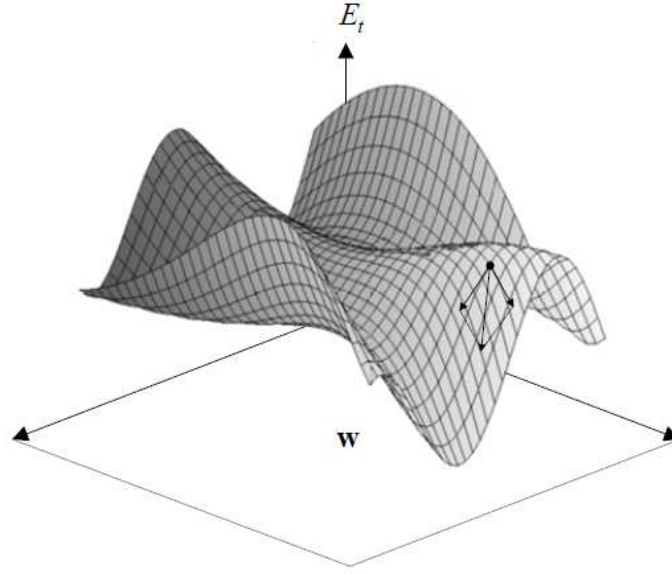


Figura 60. Aproximación numérica de la búsqueda del mínimo global en el “landscape error”.

El vector de umbrales y pesos óptimos w , se representa por un mínimo global en dicho error. Dado que no podemos calcular este mínimo analíticamente, tenemos que usar una aproximación numérica. Hay varios algoritmos de optimización numérica, tales como el *método de Newton* o el *descenso de gradiente*. Usamos el método del descenso de gradiente para encontrar el mínimo global en el error de landscape. El único paso del algoritmo iterativo se muestra a continuación:

$${}^{k+1}\vartheta_i^{(l)} = {}^k\vartheta_i^{(l)} - \lambda \frac{\partial E}{\partial {}^k\vartheta_i^{(l)}} \cdot \mu \cdot ({}^k\vartheta_i^{(l)} - {}^{k-1}\vartheta_i^{(l)})$$

$${}^{k+1}w_{i,j}^{(l)} = {}^kw_{i,j}^{(l)} - \lambda \frac{\partial E}{\partial {}^kw_{i,j}^{(l)}} \cdot \mu \cdot ({}^kw_{i,j}^{(l)} - {}^{k-1}w_{i,j}^{(l)})$$

donde ${}^kw_{i,j}^{(l)}$ es la ponderación de la conexión entre la neurona i de la capa l y la neurona j en la capa l calculada en un paso k del proceso iterativo.

La velocidad de convergencia está representada por el parámetro λ . Un valor demasiado pequeño del parámetro λ hace que la convergencia sea excesivamente lenta. Un valor demasiado grande de λ rompe la monotonía de la convergencia. La variable μ es un valor de momento o impulso, que impide que el algoritmo quede atrapado en mínimos locales.

Nota: La notación $g \left[\frac{\partial E}{\partial \vartheta_i^{(l)}} \right]$ significa “el componente $\frac{\partial E}{\partial \vartheta_i^{(l)}}$ del vector (o gradiente). La notación $\frac{\partial E}{\partial \vartheta_i^{(l)}}$ es la derivada parcial de la función error E por el valor umbral $\vartheta_i^{(l)}$. Similarmente, $\frac{\partial E}{\partial w_{i,j}^{(l)}}$ es una derivada parcial de la función E por el valor de ponderación $w_{i,j}^{(l)}$.

7.5 Análisis heurístico de caracteres

El algoritmo de segmentación descrito en el capítulo 5 a veces puede detectar elementos redundantes, que no se corresponden con caracteres apropiados. La forma de estos elementos después de la normalización es en ocasiones similar a la forma de los caracteres. Debido a esto, estos elementos no son fiablemente separables por métodos OCR tradicionales, pero varían en tamaño así como en contraste, brillo o matiz. Dado que los métodos de extracción de características descritos en el capítulo 6 no tienen en cuenta estas propiedades, existe una necesidad de utilizar análisis heurísticos adicionales para filtrar los no-caracteres. El análisis espera que todos los elementos tengan propiedades similares. Elementos con propiedades considerablemente diferentes se tratan como no válidos y se excluyen del proceso de reconocimiento.

El análisis consiste en dos fases. La *primera fase* trata sobre las estadísticas de brillo y contraste de los caracteres segmentados. Los caracteres entonces son normalizados y procesados por el algoritmo de extracción de piezas.

Dado que la extracción de piezas y la normalización del brillo alteran las propiedades estadísticas de la segmentación de caracteres, es necesario proceder a la primera fase del análisis antes de la aplicación del algoritmo de extracción de piezas.

Además, las alturas de los segmentos detectados son los mismos para todos los caracteres. Debido a esto, es necesario proceder al análisis de las dimensiones después de la aplicación del algoritmo de extracción de piezas. El algoritmo de extracción de piezas elimina el relleno blanco que rodea al carácter.

Con respecto a las limitaciones anteriores, la secuencia de pasos se desglosa a continuación:

1. Segmentar la placa.

2. Analizar el brillo y el contraste de los segmentos, y excluir los fallidos.
3. Aplicar en los segmentos el algoritmo de extracción de piezas.
4. Analizar las dimensiones de los segmentos y excluir los no permitidos.

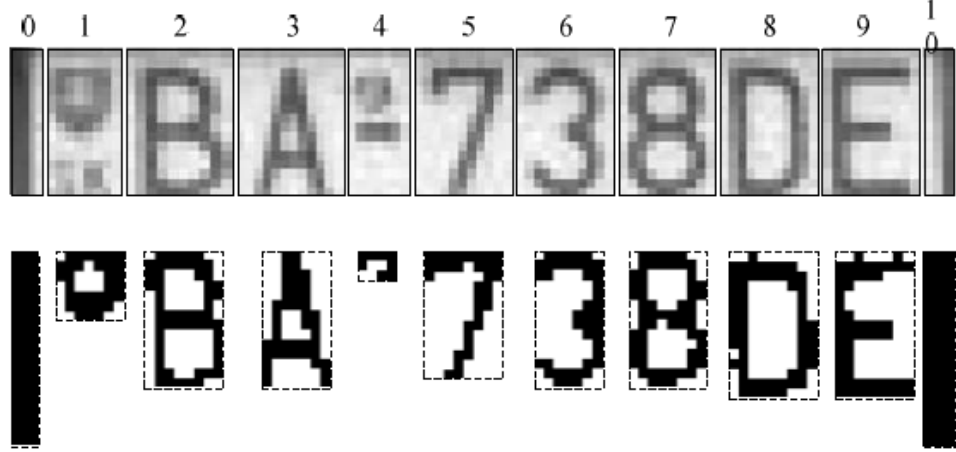


Figura 61. Caracteres segmentados antes (a) y después (b) de aplicar el algoritmo de extracción.

Si asumimos que no hay grandes diferencias de brillo y contraste en los segmentos, podemos excluir los que difieran considerablemente de la media. Sea el segmento i de la placa definida por una función discreta $f_i(x, y)$, donde w_i y h_i son las dimensiones del elemento, y en el que a continuación definimos las propiedades estadísticas de un elemento.

El *brillo global* de cada segmento se define como la media de los brillos de los pixeles individuales:

$$p_b^{(i)} = \sum_{x=0}^{w_i} \sum_{y=0}^{h_i} f(x, y)$$

El *contraste global* del segmento i se define como una desviación estándar del brillo de los pixeles individuales:

$$p_c^{(i)} = \sqrt{\frac{\sum_{x=0}^{w_i} \sum_{y=0}^{h_i} (p_b^{(i)} - f(x, y))^2}{w_i \cdot h_i}}$$

La función $f(x, y)$ representa solamente una intensidad de escala de grises, pero el adicional análisis heurístico de colores puede ser involucrado para

mejorar el proceso de reconocimiento. Este análisis separa carácter y no caracteres en base al color. Si la imagen capturada está representada un modelo de color HSV, podremos calcular directamente el tono y saturación global de los segmentos como un promedio del tono y la saturación de los pixeles individuales:

$$p_h^{(i)} = \sum_{x=0}^{w_i} \sum_{y=0}^{h_i} h(x, y); \quad p_s^{(i)} = \sum_{x=0}^{w_i} \sum_{y=0}^{h_i} s(x, y)$$

donde $h(x, y)$ y $s(x, y)$ es el tono y la saturación respectivamente, de cierto pixel en el modelo de color HSV.

Si la imagen capturada está representada por un modelo de color RGB, será necesario transformarla al modelo HSV primeramente.

Para determinar la validez del elemento, calculamos el valor medio de una propiedad elegida sobre todos los elementos. Por ejemplo, el brillo medio se calcula como $\overline{p_b} = \sum_{i=0}^{n-1} p_b^{(i)}$, donde n es el número de elementos. El elemento i se considera como válido, si el brillo global $p_b^{(i)}$ no difiere más del 16% del brillo promedio $\overline{p_b}$. Los valores umbrales de las propiedades individuales han sido calibradas de la siguiente manera:

| | | | |
|---------------------|--|-----------------------------|--|
| <i>Brillo (BRI)</i> | $\frac{p_b^{(i)} - \overline{p_b}}{\overline{p_b}} < 0.16$ | <i>Contraste (CON)</i> | $\frac{p_c^{(i)} - \overline{p_c}}{\overline{p_c}} < 0.1$ |
| <i>Tono (HUE)</i> | $\frac{p_h^{(i)} - \overline{p_h}}{\overline{p_h}} < 0.145$ | <i>Saturación (SAT)</i> | $\frac{p_s^{(i)} - \overline{p_s}}{\overline{p_s}} < 0.24$ |
| <i>Altura (HEI)</i> | $\left \frac{h_i - \overline{h}}{\overline{h}} \right < 0.2$ | <i>Anchura/Altura (WHR)</i> | $0.1 < \frac{w_i}{h_i} < 0.92$ |

Si el segmento incumple al menos una de las restricciones anteriores, se considera como no válida y se excluye del proceso de reconocimiento. La tabla 4 contiene propiedades de los elementos de la figura 61. De acuerdo con esta tabla, los elementos 0 y 10 han sido rechazados debido a una rara relación anchura / altura; y los elementos 1 y 4, debido a una pequeña altura.

Ni que decir tiene que este análisis heurístico, es perfectamente aplicable para matrículas de cualquier país. Lo que es aconsejable, configurar de manera adecuada este proceso, dependiendo el tipo y zona de la placa a analizar, para obtener así unos resultados complemente óptimos y fiables. A esto es a lo que se le llama en la jerga del reconocimiento de caracteres como “entrenar” un OCR.


| | i | BRI | CON | HUE | SAT | HEI | WHR | Restricciones violadas |
|---|----|-------|-------|-------|-------|--------|-------|------------------------|
|  | 0 | 0.247 | 0.038 | 0.152 | 0.236 | 0.189 | 0.093 | BRI, HUE, WHR |
| | 1 | 0.034 | 0.096 | 0.181 | 0.134 | -0.554 | 0.833 | HUE, HEI |
| | 2 | 0.002 | 0.018 | 0.030 | 0.038 | 0.040 | 0.642 | |
| | 3 | 0.084 | 0.012 | 0.003 | 0.061 | 0.189 | 0.625 | |
| | 4 | 0.001 | 0.003 | 0.021 | 0.059 | -0.777 | 1.666 | HEI, WHR |
| | 5 | 0.117 | 0.016 | 0.002 | 0.063 | 0.189 | 0.625 | |
| | 6 | 0.063 | 0.016 | 0.007 | 0.056 | 0.189 | 0.562 | |
| | 7 | 0.025 | 0.011 | 0.025 | 0.028 | 0.114 | 0.533 | |
| | 8 | 0.019 | 0.025 | 0.012 | 0.034 | 0.114 | 0.600 | |
| | 9 | 0.019 | 0.048 | 0.009 | 0.045 | 0.114 | 0.533 | |
| | 10 | 0.062 | 0.009 | 0.041 | 0.018 | 0.189 | 0.095 | WHR |

Tabla 4. Propiedades de los segmentos de la figura 61.

8

Análisis sintáctico de la placa

8.1 Principios y algoritmos

En algunas situaciones en las que el mecanismo de reconocimiento falla, hay una posibilidad de detectar el fallo a través de un análisis sintáctico de la placa reconocida. Si tenemos reglas específicas de cada país para la placa, podemos evaluar la validez de esa placa hacia estas reglas. La corrección sintáctica automática de la matrículas puede incrementar la capacidad de reconocimiento del sistema ANPR completo.

Por ejemplo, si el software de reconocimiento se confunde entre los caracteres “8” y “B”, la decisión final puede hacerse de acuerdo al patrón sintáctico. Si el patrón solamente permite dígitos para esa posición, el carácter “8” se utilizará en vez de la letra “B”. Este proceso también suele formar parte del proceso de “entrenamiento de un OCR”.

Otro buen ejemplo, es la decisión entre el dígito “0” y el carácter “O”. La gran pequeña diferencias entre estos caracteres, hacen su reconocimiento extremadamente difícil, y en muchos casos imposible.

8.1.1 Reconocimiento del carácter y su coste

En la mayoría de los casos, los caracteres son reconocidos por redes neurales o neuronales. Cada neurona en una capa de salida típicamente representa un carácter. Sea $y = (y_0, \dots, y_9, y_A, \dots, y_Z)$ un vector de las actividades de salida. Si hay 36 caracteres en el alfabeto, el vector y también tendrá 36 dimensiones.

Sea y_i el componente i del vector y . Por tanto, y_i significa qué cantidad o magnitud se corresponde el carácter de entrada con el carácter i del alfabeto, que se representa por este componente. El carácter reconocido χ se representa por el componente más grande del vector y :

$$\chi = chr\left(\max_{0 \leq i \leq Z} \{y_i\}\right)$$

donde $chr(y_i)$ es el carácter, que se presenta por el componente i del vector y .

Sea $y^{(s)}$ un vector, y ordenado de forma descendente de acuerdo con los valores de los componentes. Por tanto, el carácter reconocido está representado por el primer componente de tal vector ordenado:

$$\chi = chr\left(y_0^{(s)}\right)$$

Cuando el proceso de reconocimiento falla, el primer componente de $y^{(S)}$ puede contener un carácter inválido, que no está acorde con el patrón de sintaxis. Entonces, es necesario utilizar el siguiente carácter válido con un coste peor.

8.1.2 Patrones sintácticos

En la práctica, los sistemas ANPR deben hacer frente a muchos diferentes tipos de matrículas. Las matrículas no son únicas, cada país o región tienen un tipo propio. Debido a esto, un sistema de reconocimiento de matrículas debe ser capaz de reconocer un tipo de matrícula, y asignar automáticamente el patrón sintáctico correcto a ésta. La asignación del patrón sintáctico correcto es un problema fundamental en el análisis sintáctico.

El patrón sintáctico es un conjunto de reglas que definen los caracteres, que pueden ser usados en una cierta posición de una matrícula. Si la matrícula \mathbf{P} es una secuencia de n caracteres alfanuméricos $P = (p^{(0)} \dots p^{(n-1)})$, entonces el patrón sintáctico P' es un n -múltiplo de conjuntos $P' = (p'^{(0)} \dots p'^{(n-1)})$, y $p'^{(i)}$ es un conjunto de todos los caracteres permitidos para la posición i en una placa.

Por ejemplo, las matrículas españolas a partir del año 2000, que son en las que nos centramos en el código elaborado para este proyecto, contienen, cuatro dígitos seguidos de tres letras. El patrón sintáctico para las matrículas españolas sería el siguiente:

$$P' = \left(\begin{array}{l} \{0,1,2,3,4,5,6,7,8,9\}, \{0,1,2,3,4,5,6,7,8,9\}, \{0,1,2,3,4,5,6,7,8,9\}, \{0,1,2,3,4,5,6,7,8,9\}, \\ \{B, C, D, F, G, H, J, K, L, M, N, P, R, S, T, V, W, X, Y, Z\}, \\ \{B, C, D, F, G, H, J, K, L, M, N, P, R, S, T, V, W, X, Y, Z\}, \\ \{B, C, D, F, G, H, J, K, L, M, N, P, R, S, T, V, W, X, Y, Z\} \end{array} \right)$$

8.1.3 Elección del patrón correcto

Si hay n patrones sintácticos $P'^{(0)} \dots P'^{(n-1)}$, tenemos que escoger el más adecuado para la matrícula \mathbf{P} . Para este propósito, definimos un costo δ para el cálculo de una similitud entre la matrícula evaluada y el correspondiente patrón sintáctico:

$$\delta(P') = |\{p^{(i)} | p^{(i)} \notin p'^{(i)}\}| + \sum_{i=0}^{n-1} \left(\frac{1}{\max_{0 \leq j \leq z} \{y_j^{(i)}\}} \cdot 10^{-2} \right)$$

donde $|\{p^{(i)} | p^{(i)} \notin p'^{(i)}\}|$ es un número de caracteres, que no coinciden con las correspondientes posiciones en el patrón sintáctico P' . Sea $y^{(i)}$ un vector de

salida para el carácter i reconocido en una placa. El mayor componente de dicho vector, $\max_{0 \leq j \leq z} \{y_j^{(i)}\}$, indica cómo de exitoso ha sido el reconocimiento de la placa. Por tanto, el valor recíproco de $\max_{0 \leq j \leq z} \{y_j^{(i)}\}$ es un costo del carácter. Otra manera de evaluar el coste es el uso del *algoritmo de Smith-Waterman* para calcular la diferencia entre la matrícula reconocida y el patrón sintáctico.

Por ejemplo, supongamos que la matrícula '1732BDY' ha sido reconocida como '17328DY', y el patrón de reconocimiento no permite un dígito en la quinta posición de una matrícula. Si el carácter '8' ha sido reconocido con una relación de similitud de 0.90 y otros caracteres con la relación 0.95, la métrica de este patrón determina lo siguiente:

$$\delta(P') = (1) + \left(\frac{10^{-2}}{0.95} + \frac{10^{-2}}{0.95} + \frac{10^{-2}}{0.95} + \frac{10^{-2}}{0.95} + \frac{10^{-2}}{0.90} + \frac{10^{-2}}{0.95} + \frac{10^{-2}}{0.95} \right)$$

Si hay un patrón que coincide exactamente con la matrícula evaluada, podremos decir que ese número ha sido correctamente reconocido, y no necesita de más correcciones. Además, no es posible detectar una matrícula errónea, si no rompe las reglas de un patrón sintáctico.

De lo contrario, será necesario corregir la placa detectada usando el patrón de menor coste δ :

$$P'^{(sel)} = \arg \min_{0 \leq i \leq n} \{\delta(P'^{(i)})\}$$

La corrección de la placa significa el reemplazo de cada carácter no válido por otro. Si el carácter $p^{(i)}$ en la posición i de la placa \mathbf{P} no coincide con el patrón seleccionado $P'^{(sel)}$, será reemplazado por el primer válido de $y^{(s)}$. El vector $y^{(s)}$ de actividades de salida ordenadas, que indican el carácter reconocido cuánto es de similar a un carácter individual del alfabeto.

El análisis heurístico de la placa segmentada puede a veces incorrectamente evaluar un no carácter como carácter. La aceptación de los no caracteres hace que la placa reconocida contenga elementos redundantes. Los caracteres redundantes pueden aparecer ocasionalmente en los lados de la placa, pero muy rara vez en el centro de la misma. Si la matrícula reconocida es más larga que el patrón de sintaxis mayor, podemos seleccionar el patrón más cercano, y colocar los caracteres o elementos redundantes de acuerdo con éste.

9

Pruebas y consideraciones finales

9.1 Elección del conjunto representativo de imágenes

Se han capturado diversas instantáneas en movimiento y de vehículos estacionados como propósito de prueba. Primeramente, el objetivo era encontrar un conjunto representativo de matrículas que fueran reconocibles por humanos. Por supuesto, un conjunto como este contendría un espectro extremadamente amplio de placas, tales como, claras y fácilmente reconocibles, así como placas degradadas por un significativo efecto de movimiento o inclinación.

Entonces, la capacidad de reconocimiento de una máquina está representada por la relación entre el número de placas, que han sido reconocidas por la máquina, y el número de placas reconocidas por un humano. Prácticamente, es imposible construir una máquina con las mismas capacidades de reconocimiento que un humano. Debido a ello, una prueba como ésta sería extremadamente difícil e inútil.

En la práctica, es más útil encontrar un conjunto representativo de matrículas, que pueden ser capturadas por una cámara ANPR. La posición de la cámara tiene un significativo efecto en la calidad de las imágenes capturadas, y un éxito en el proceso de reconocimiento conjunto. La posición de la cámara hacia el carril puede llevar a un mejor conjunto instantáneas posibles. En algunas situaciones, podemos evitar conseguir instantáneas desviadas o distorsionadas con un posicionamiento adecuado de la cámara. A veces, esto es más inteligente y eficaz que un desarrollo de los robustos mecanismos de enderezado.

Sea S un conjunto representativo de todas las instantáneas, que pueden ser capturadas por la cámara ANPR. Algunas de la instantáneas de este conjunto pueden estar borrosas, otras pueden ser demasiado pequeñas, grandes, distorsionadas o deformadas. Debido a esto, se ha dividido el conjunto en los siguientes subconjuntos:

$$S = S_c \cup S_b \cup S_d \cup S_e \cup S_p$$

donde S_c es un subconjunto de placas nítidas o sin distorsión o degradación, S_d es un subconjunto de matrículas degradadas o distorsionadas, S_e es un subconjunto que tiene un difícil entorno circundante para su reconocimiento, y S_p contiene placas con caracteres pequeños.



Figura 62. Ejemplos de imagen del conjunto de matrículas (a) claras (b) con pequeños o caracteres borrosos (c) matrículas distorsionadas (d) placas con entornos conflictivos o sombras parciales.

9.2 Evaluación de la exactitud de una matrícula

Las matrículas reconocidas por una máquina pueden a veces diferir de las correctas. Debido a ello, es necesario definir unas fórmulas y reglas, que serán usadas para evaluar el grado de la exactitud de la placa.

Sea \mathbf{P} una matrícula, y $\mathbf{S} = \{\mathbf{P}^{(0)}, \dots, \mathbf{P}^{(n-1)}\}$ un conjunto de todas las matrículas probadas. Por tanto, la tasa de reconocimiento $R(\mathbf{S})$ del sistema ANPR probado en el conjunto \mathbf{S} se calcula como:

$$R(\mathbf{S}) = \frac{1}{n} \sum_{i=0}^{n-1} s(\mathbf{P}^{(i)})$$

donde n es una cardinalidad del conjunto \mathbf{S} , y $s(P^{(i)})$ es un resultado correcto de la placa \mathbf{P} . La *puntuación de la exactitud* es un valor, que expresa cómo de exitoso ha sido el reconocimiento de la placa.

Ahora, la cuestión es cómo definir la exactitud de placas individuales. Hay tres diferentes enfoques de cómo evaluarlas: puntuación binaria y puntuación ponderada y porcentaje.

9.2.1 Puntuación binaria

Digamos que la matrícula \mathbf{P} es una secuencia de n caracteres alfanuméricos $\mathbf{S} = (p^{(0)}, \dots, p^{(n-1)})$, si $P^{(r)}$ es la matrícula reconocida por una máquina, y $P^{(c)}$ es la correcta, la puntuación binaria s_b de la placa $P^{(r)}$ se evalúa de la siguiente manera:

$$s_b(P^{(r)}) = \begin{cases} 0 & \text{si } P^{(r)} \neq P^{(c)} \\ 1 & \text{si } P^{(r)} = P^{(c)} \end{cases}$$

Dos matrículas son iguales, si todos los caracteres en las correspondientes posiciones son iguales:

$$(P^{(r)} = P^{(c)}) \leftrightarrow (\forall i \forall j: p_i^{(r)} \neq p_j^{(r)} \rightarrow i \neq j)$$

donde $p_i^{(r)}$ es el carácter i de la matrícula $P^{(r)}$.

9.2.2 Puntuación ponderada

Si $P^{(r)}$ es una matrícula reconocida por una máquina, y $P^{(c)}$ es la correcta, entonces la puntuación ponderada s_w de la matrícula $P^{(r)}$ está dada como:

$$s_w(P^{(r)}) = \frac{|\{p_i^{(r)} | p_i^{(r)} = p_i^{(c)}\}|}{|\{p_i^{(r)}\}|} = \frac{m}{n}$$

Donde m es el número correcto de caracteres identificados, y n es el número de todos los caracteres de la matrícula.

Por ejemplo, si la matrícula "1234BCD" ha sido reconocida como "12348CD", la puntuación ponderada exacta para esta placa es del 0.85, pero la puntuación binaria sería 0.

9.2.3 Porcentaje

Se trata del tipo de puntuación más clásico, común y efectivo, que se usa para observar rápidamente los resultados de un proceso. Teniendo en cuenta las variables de los dos procesos anteriores:

$$s_p(p^{(r)}) = \frac{p_i^{(r)}}{p_i^{(c)}} \cdot 100$$

9.3 Resultados

La siguiente tabla nos muestra las tasas de reconocimiento, que han sido logradas durante las pruebas en varios conjuntos de matrículas. Han sido utilizadas en el proceso 125 matrículas españolas con dimensiones 600x400 de media, para llevar a cabo los experimentos.

| | Número total de placas | Número total de caracteres acertados | Porcentaje | Puntuación ponderada |
|-------------------|------------------------|--------------------------------------|------------|----------------------|
| Placas nítidas | 53 | 359/371 | 96,77 | 89,29 |
| Placas difusas | 42 | 265/294 | 90,14 | 65,37 |
| Placas deformadas | 30 | 175/210 | 83,33 | 54,82 |
| Media | 125 | 799/875 | 91,31 | 71,13 |

Tabla 5. Tasas de reconocimiento del sistema ANPR según el tipo de matrícula.

| Unidades del sistema LPR | Número de aciertos | Porcentaje de aciertos (%) |
|--------------------------|--------------------|----------------------------|
| Localización de la placa | 107/125 | 85,6 |
| Segmentación | 115/125 | 92 |
| Extracción | 118/125 | 94,4 |
| Reconocimiento | 121/125 | 96,8 |
| Media | 115/125 | 92,01 |

Tabla 6. Tasa de reconocimiento del sistema LPR global.

Según los resultados, este sistema da una buena respuesta en placas nítidas, pero para placas deformadas y con entornos sintácticamente conflictivos causan una significativa degradación en las capacidades de reconocimiento.

Con respecto a los componentes del sistema LPR, vemos que tiene un excelente comportamiento en la extracción y el reconocimiento de los caracteres, así como unos buenos resultados en el proceso de segmentación. En cuanto a la localización de la placa, el sistema se comporta más deficientemente, ya que se trata de la parte del sistema más compleja y sensible a los errores.

10 Conclusiones y líneas futuras

A lo largo del presente proyecto, hemos desarrollado tanto técnicamente como matemáticamente las distintas partes del software de reconocimiento de un sistema ANPR, al cual también hemos apoyado con una parte práctica, desarrollando un código en MATLAB (incluido en el CD adjunto), donde hemos ido comprobando las funcionalidades y propiedades de los diferentes algoritmos relacionados con los apartados expuestos.

9.4 Cumplimiento de objetivos y conclusiones

A continuación procederemos a analizar el cumplimiento de los objetivos teóricos iniciales que se propusieron antes de la realización del proyecto:

- Se ha conocido y estudiado la historia y características de la matriculación de los vehículos en nuestro país, para poder abordar eficazmente la problemática propuesta inicialmente.
- Se han desarrollado completamente, tanto las partes físicas y técnicas de un sistema ANPR común y explicado su funcionamiento a nivel global. La parte de software, es la que se ha elegido para desarrollar en este proyecto y realizar un entendimiento de su proceso.
- En el afán de explicar y desarrollar su parte logarítmica, hemos dividido y clasificado sus diferentes pasos o etapas de forma casi jerárquica, aunque se ha comprobado que el cambio de posición de alguno de estos procesos, no tiene porqué modificar su resultado, aunque sí su carga de procesado o volumen.
- Se ha desarrollado en profundidad el apartado de reconocimiento de áreas en una imagen, explicando y exponiendo diversos pasos o métodos como la detección de bordes, las proyecciones de imagen, análisis estadísticos y heurísticas, así como un proceso de rotación y corrección para poder obtener un matrícula perfectamente rotada y en lo mejor posible distinguible. Hemos podido comprobar que esta fase es la más compleja, sensible y costosa del sistema software LPR, ya que de ésta depende casi el 50% del éxito de que reconozca una matrícula, siendo la piedra angular del sistema, y como hemos podido comprobar en el desarrollo y resultados obtenidos en el código.

- Se ha estudiado los principios de segmentación de la placa, completamente necesario para obtener sus segmentos característicos y así poder obtener más adelante los caracteres de la matrícula que estamos buscando o saber de qué país o región es la misma.
- Una vez segmentada la placa, hemos explicado el proceso de extracción de características y normalización de los segmentos. Hemos llevado a cabo procesos como la normalización del contraste y brillo, de sus dimensiones y la extracción de sus características, para poder compararlas con una determinadas de referencias y baremos previamente establecidos en una base de datos.
- A continuación se ha expuesto un proceso elemental e indispensable de los sistemas ANPR, el reconocimiento de los caracteres u OCR. En éste apartado hemos vistos los distintos métodos con los cuales se lleva a cabo el reconocimiento de los caracteres buscados, como las redes neuronales, que simulan el comportamiento del cerebro humano, y el análisis heurístico de los resultados obtenidos para ser más certeros en lo que estamos buscando.
- Se han desarrollado los principios y algoritmos para un análisis sintáctico de la placa obtenida, con los que asegurarnos que el tipo de carácter (letra o número) es correcto por su sintaxis y posición previamente establecida. Este aspecto siempre va a depender del país y la región de la matrícula que estemos analizando, ya que estos tienen una estructura de símbolos y caracteres predefinida.
- Por último, se han analizados los resultados de aplicar, la mayoría de los conceptos expuestos en el proyecto, y así poder sacar conclusiones del comportamiento y el funcionamiento de los sistemas ANPR.

Con respecto a la parte práctica del proyecto que se ha realizado en MATLAB para justificar la parte teórica del proyecto, se han desarrollado los siguientes procesos:

- Desde un principio se tuvo la idea que crear un algoritmo rápido de reconocimiento de matrículas y caracteres. En este se han combinado algoritmos descritos en el proyecto y otros propiamente diseñados para dar un algoritmo liviano, rápido y eficaz.
- Primeramente se creó un algoritmo por transformación proyectiva de coordenadas, en el cual se definen unos parámetros de entrada y

en el que se produce un comportamiento la mayoría de las veces fiables.

- Después se aplican unos filtros de detección de bordes Canny de rango 0.5, con el cual se definirán los bordes de la imagen. La utilización de este rango se debe a que el algoritmo solo mostrará bordes gruesos (como los de las matrículas) y desechará los finos (como bordes de hojas en el paisaje, etc). Si el rango es cada vez más alto buscará bordes aún más gruesos, dejando bordes inconexos, caso que no queremos por ningún medio, ya que vamos a buscar piezas unidas.
- A continuación se aplica un algoritmos de dilatación de píxeles, para que los bordes se junten, si aún no lo estaban, es decir crear especies de "lagunas" o "lazos" de píxeles. Realizamos la operación anterior para rellenar esas estructuras, por lo que nos quedarán varios segmentos rellenos de píxeles blancos.
- Lo que tenemos son trozos blancos en una imagen negra. Buscaremos el área rellena más grande, que en la mayoría de los casos se trata de la matrícula, por lo que aislaremos los demás huecos.
- Una vez aislada la parte mayor, la utilizamos como plantilla para realizar un corte en la imagen original, con lo que solo tendremos la imagen de la placa.
- Pasaremos a escala de grises la placa y realizaremos una umbralización global para un valor de umbral de brillo de 125.
- Con la placa umbralizada procedemos a segmentar la imagen donde obtendremos 7 segmentos justos o más. Si hay más de siete, los restantes serán redundantes, por lo que hay que aplicar un buen criterio, para no perder ningún carácter. Se ha diseñado para ello un algoritmo que coja los 7 primeros segmentos de mayor área, con lo que nos quedará una imagen binaria con los caracteres obtenidos.
- Finalmente se aplica un OCR entrenado (con las propiedades características de la matrícula española) en el que buscará los caracteres de la imagen y los comparará con un vector de letras y números, y finalmente los mostrará en un documento de texto.

Visto esto, podemos afirmar que se han cumplido satisfactoriamente todos los objetivos pensados e ideados para este proyecto, además de añadir diseños

logarítmicos propios e inteligentes para satisfacer el estudio de reconocimiento de caracteres.

9.5 Líneas futuras

Una vez expuestas las conclusiones y resultados del proyecto, se podrían desarrollar diversas ideas en un futuro, las cuales perfeccionen y enriquezcan las funciones desarrolladas para este fin, tales como:

- Ampliar el reconocimiento de matrículas españolas, no solo a las actuales, sino que también hacia placas antiguas y especial como las mostradas en el apartado 2.
- Ampliar el reconocimiento hacia matrículas extranjeras. Un sistema de reconocimiento ideal, sería el capaz de reconocer matrículas de cualquier país, a partir de su sintaxis y estructura morfológica.
- Automatizar y optimizar el proceso de detección y rotación de áreas. Como se ha visto en la teoría y en la práctica, se trata de la parte más compleja y más fallos da en el sistema ANPR. Sería ideal que el propio sistema detectara y rotara la matrícula a partir de la transformada de Hough, de una manera más rápida y automatizada que la realizada en el proyecto.
- Realización de un algoritmo de umbralización adaptivo. Un sistema ANPR ideal, aparte de reconocer nacionalidad de la matrícula, es que sea capaz de tratar y reconocer cualquier tipo de matrícula, a pesar de los problemas que presente la imagen. Este algoritmo sería ideal para placas en las que una parte de la matrícula se presenta oscura, como las que son ensombrecidas por la barra de un remolque trasero o por las luces de este mismo mal colocadas, etc, los cuales crean un efecto indeseable para la normalización de los elementos de la palca.
- Implementar un sistema automático de lectura de matrículas. La ideas es introducir un número determinado de matrículas en un array o vector , que sea procesada por el programa y a su vez crear otra matriz de salida con los caracteres organizados en matriculas, para su posterior procesado. Esto supondría una mayor rapidez en la obtención de las matrículas, así como la agilidad de tratamiento de los datos en sucesos posteriores.

Bibliografía

- [1] David Uríos Mondejar, "Sitio de las Matrículas Españolas", 2001-20012.
<http://www.sme-matriculas.es/>
- [2] E. Rajoy, "anexo XVIII del Reglamento General de Vehículos, aprobado por Real Decreto 2822/1998", 15 de Septiembre de 2000.
http://noticias.juridicas.com/base_datos/Admin/o150900-mp.html#dfu
- [3] "Reconocimiento automático de matrículas", Wikipedia, La Enciclopedia Libre,
http://es.wikipedia.org/wiki/Reconocimiento_autom%C3%A1tico_de_matr%C3%ADculas
- [4] Adaptation Recognition Hungary, "Automatic License Plate Recognition", 2012, <http://www.platerecognition.info/index.htm>
- [5] Chris Solomon, Toby Breckon, "Fundamentals of Digital Image Processing. A Practical Approach with Examples in Matlab", John Wiley & Sons Ltd., 2011.
- [6] González, R.C., Wintz, P. , "Procesamiento digital de imágenes". Addison-Wesley, 1996.
- [7] Jain, R., Kasturi, R., Schunck, B., "Machine Vision" , McGraw-Hill International Editions, 1995.
- [8] Huaifeng Zhang, Wenjing Jia, Xiangjian He, Qiang Wu, "A Fast Algorithm for License Plate Detection in Various Conditions", IEEE International Conference on Systems, Man, and Cybernetics, 2006.
- [9] Shishir Kumar, Shashank Awargal, Kumar Saurab, "License Plate Recognition System for Indian Vehicles", International Journal of Information Technology and Knowledge Management., Julio-Diciembre 2008.
- [10] Carlos Parra Ramos, David Regajo Rodriguez, "Reconocimiento Automático de Matrículas", Universidad Calos III de Madrid, 2006.
- [11] Bishop, C.M., "Neuronal Networks for Pattern Recognition". Oxford University Press, 1995.

- [12] Ondrej Martinsky, "Algorithmic and Mathematical Principles of Automatic Number Plate Recognition Systems", Brno University of Technology, 2006.
- [13] Rodrigo Abal, Raúl Medeglia, Nicolás Pebet, "Reconocimiento Automático de Matrículas", Universidad de la República, Uruguay, 2003.
- [14] Erik Bergenudd, "Low-Cost Real-Time License Plate Recognition for a Vehicle PC", KTH Electrical Engineering, Suecia, 2006.
- [15] Serkan Ozbay, Ergun Ercelebi, "Automatic Vehicle Identification by Plate Recognition", World Academy of Science, Engineering and Technology, 2005.
- [16] Ali Tahir, Hafiz Adnan Habib, M. Fahad Khan, "License Plate Recognition Algorithm for Pakistani License Plates", University of Engineering & Technology Taxila, Pakistan, 2007.
- [17] B.D. Acosta, "Experiments in image segmentation for automatic US license plate recognition", M.Sc. thesis, Department of Computer Science, Faculty of the Virginia Polytechnic Institute and State University, 2004.
- [18] Tom Jongsma, "Image processing designtask: license plate localization", Educational program MSc Twente, Holanda, 2008.
- [19] J. F. Canny, "A computational approach to edge detection", IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-698. 1986.
- [20] Antonio Ignacio Betancor Pérez, "Sistema De Reconocimiento de Matrículas Basado en Visión Artificial para Control de Acceso", Universidad Politécnica de Cartagena, 2008.
- [21] Amr Badr, Mohamed M. Abdelwahab, Ahmed M. Thabet, and Ahmed M. Abdelsadek, "Automatic Number Plate Recognition System", Annals of the University of Craiova, 2011.
- [22] Henrik Hansen, Anders Wang Kristensen, Morten Porsborg Køhler, Allan Weber Mikkelsen, Jens Mejdahl Pedersen and Michael Trangeled, "Automatic recognition of license plates", Aalborg Universitet, 2002.
- [23] Christina Olsén, "Image Analysis: OCR-classification", Umeå University, 2004.
- [24] Jamey Brown, Alantria Harris, Matthew Noury, Amit Patel, M. John Rafferty, "Plate Watcher", Georgia Institute of Technology, 2002.

[25] Kerry Widder, "Automatic License Plate Location Using Template Matching", University of Wisconsin - Madison, 2004.

[26] N. Ketelaars, "Final Project: Automated License Plate Recognition," AIME Magazine, 2002.

[27] Mathworks, "MATLAB Central", 2012. www.mathworks.es

[28] Danilo Simón Zorita, "Adquisición de Firmas Off-line", Universidad Politécnica de Madrid, 2010.

[29] Federico Debandi, Miriam Redi, "El reconocimiento de textos", Università degli Studi di Torino, 2009.

Apéndice I:

Notaciones y

símbolos

matemáticos

Símbolos y operaciones lógicas

$p \oplus q$ Unión excluyente (o p o q)

$p \cap q$ Intersección (p y q)

$p \cup q$ Unión (p o q)

$\neg p$ Negación (no p)

Definición matemática de la imagen

$f(x, y)$ x e y son las coordenadas espaciales de una imagen, y f es el nivel de brillo en ese punto. Esta función siempre va a ser discreta. $x \in \mathbb{N} \cap y \in \mathbb{N}$, donde \mathbb{N} engloba todos los números naturales incluyendo cero.

$f(p)$ Intensidad de luz en un punto p . $f(p) = f(x, y)$, donde $p = [x, y]$

Píxeles vecinos

$p_1 \ddot{N}_4 p_2$ El píxel p_1 tiene colindancia o conexión de 4 píxeles con el píxel p_2 (y viceversa)

$p_1 \ddot{N}_8 p_2$ El píxel p_1 tiene colindancia o conexión de 8 píxeles con el píxel p_2 (y viceversa)

Producto de convolución

$a(x) * b(x)$ Convolución discreta de las señales $a(x)$ y $b(x)$

$a(x) \tilde{*} b(x)$ Convolución periódica discreta de las señales $a(x)$ y $b(x)$

Vectores y conjuntos

$M[m, n]$ Elemento situado en la columna m y fila n de la matriz M

$\max A$ Elemento de máximo valor del conjunto A . El alcance de los elementos puede ser especificado por condiciones adicionales

$\min A$ Elemento de valor mínimo del conjunto A .

$\text{media } A$ Valor medio de los elementos contenidos en el conjunto A

$\text{mediana } A$ Mediana de los elementos contenidos en el conjunto A

$|A|$ Número de elementos contenidos en el conjunto A

X Los vectores se indican con negrita

x_i Elemento de un vector, donde i es el índice del mismo (empezando por cero) tal que $i \in 0 \dots n - 1$.

| | |
|-----------|---|
| $X[a]$ | <i>Elemento a del vector x. Por ejemplo, el vector x puede contener los elementos a, b, c y d, tal que $x=(a,b,c,d)$.</i> |
| $X^{(i)}$ | <i>Si hay más de un vector denotado como X, estos se diferencian por su índice i. El índice superior i no significa el enesimo elemento del vector.</i> |

Intervalos

$a < x < b$ *x se encuentra en el intervalo entre a y b. Esta notación es usada cuando x es la coordenada espacial en una imagen (tanto en el tiempo discreto como en el continuo)*

$x \in a...b$ *Significa lo mismo que la notación anterior solo que ésta es válida solamente para el tiempo discreto.*

Cuantificadores

| | |
|------------------|--------------------------------------|
| $\exists x$ | <i>Existe al menos para una x</i> |
| $\exists! x$ | <i>Existe exactamente para una x</i> |
| $\exists^n x$ | <i>Existe exactamente para n x</i> |
| $\neg \exists x$ | <i>No existe para x</i> |
| $\forall x$ | <i>Para todo x</i> |

Redondeos

| | |
|---------------------|---|
| $\lfloor x \rfloor$ | <i>Redondeo de x al entero inferior más cercano</i> |
| $\lceil x \rceil$ | <i>Redondeo de x al entero superior más cercano</i> |

Apéndice II:

Contenido del CD

El CD adjunto al proyecto contiene las siguientes ficheros:

- *proyecto.pdf*: Contiene la identificación de los datos del proyecto.
- *memoria.pdf/.doc*: Contiene la memoria elaborada para este proyecto
- *resumen.pdf*: Se trata de un resumen del proyecto en 300-600 palabras.
- *codigo_MATLAB*: En esta carpeta se incluye el código elaborado en MATLAB para la parte práctica del proyecto. Contiene las siguientes subcarpetas y funciones:
 - *letters_numbers*: contiene las imágenes de los caracteres que utiliza para crear los vectores de caracteres.
 - *matriculas*: diversos ejemplos de matrículas utilizadas por el algoritmo para comprobar la efectividad del sistema de reconocimiento elaborado.
 - *principal.m*: se trata de la función principal del programa donde llama a diferentes funciones y obtener el resultado final.
 - *create_templates_letr.m*: función que se encarga de crear el vector de letras a partir de los caracteres de la carpeta *letters_numbers*.
 - *create_templates_num.m*: función que se encarga de crear el vector de números a partir de los caracteres de la carpeta *letters_numbers*.
 - *templates*: vector matemático creado por *create_templates_letr.m*
 - *templates_num*: vector matemático creado por *create_templates_num.m*
 - *obtiene_caracter.m*: función utilizada para normalizar las dimensiones de los caracteres.
 - *prepararfotomatrícula.m*: función que realiza la transformación proyectiva de una imagen.
 - *OCR.m*: función encargada de realizar el reconocimiento óptico de caracteres. Está formada por las siguientes subfunciones:
 - *lines_crop.m*: se encarga de obtener las líneas o segmentos existentes en la imagen.

- *letter_crop.m*: se encarga de obtener los caracteres existentes en la imagen.
- *read_letter_2.m*: lee el carácter y escribe en el .txt el número correspondiente.
- *rear_letter_3.m*: lee el carácter y escribe en el .txt la letra correspondiente.
- *umbralizacion.m*: se encarga de umbralizar una placa en escala de grises.
- *text.txt*: matrícula en documento de texto, de la última matrícula reconocida.